International Journal of Advanced Research Foundation
Website: www.ijarf.com, Volume 1, Issue 2, November 2014)

Design and Implementation of PAB Based On Chip Permutation Network for Multiprocessor SOC

Mr. K.S.V.Raghu Babu
M.Tech scholar, VLSI
Aurora's Scientific, Technological and Research Academy
Hyderabad, India
Babu.raghu786@gmail.com

Mrs. D.Kavitha
Associate Professor, Dept.of E.C.E
Aurora's Scientific, Technological and Research Academy
Hyderabad, India

Mrs. V.Rajitha Rani
Sr.Assistant Professor, Dept.of E.C.E
Aurora's Scientific, Technological and Research Academy
Hyderabad,India

***Abstract*— This paper presents the design of a novel OCP network to support guaranteed traffic permutation in multiprocessor SOC applications. The proposed network utilizes 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 existing system having only fixed priority logic scheme for dynamic path set up .In this paper we proposed a new PAB based priority logic to rectify the drawbacks in previous arbiter system in proposed OCP network. The PAB contains F-Priority, RR-Priority, D-Priority logics. This circuit-switching approach offers a guarantee of permuted data and its compact overhead enables the benefit of stacking multiple net-works. The proposed on chip network improves the efficiency. Finally implemented the design using Xilinx ISE 12.1 software on FPGA Spartan 3E family kit, NEXYS 2 board and showed the synthesis result and power result. The Proposed system OCP network with PAB improves the power, delay efficiency and improves the data efficiency**.

***Index Terms*— SOC, PAB, OCP, FPGA, Circuit Switching, Dynamic Path setup, F-Priority, RR-Priority, D-Priority.**

I. INTRODUCTION

For applications of parallel processing, scientific computing, and so on, In a present trend of multiprocessor system on chip (MPSoC) design are interconnected with on-chip networks is currently emerged[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 among the important traffic classes exhibited from on-chip multiprocessing applications [7],[8]. Standard permutations of traffic occur in general-purpose MPSoCs, for example, like polynomial, sorting, and fast Fourier transform (FFT) computations cause shuffled permutation, whereas matrix transposes or corner-turn operations exhibit transpose permutation [6]. Recently, application specific MPSoCs targeting flexible Turbo/LDPC decoding had developed, and they exhibit arbitrary and concurrent traffic permutations due to multi-mode and multi-standard feature [3-5]. In addition to that many of the MPSoC applications (e.g., Turbo/LDPC decoding [3-5]) compute in real-time, therefore, guaranteeing throughput is critical for such permutation traffics. Most on-chip networks in practice are general-purpose and use routing algorithms such as minimal adaptive routing and dimension-ordered routing. To support permutation traffic patterns, on-chip permutation networks using application-aware routings are needed to achieve better performance compared to the general-purpose networks [8].

These application-aware routings are configured before running the applications and can be implemented as source routing or distributed routing. But such application-aware routings cannot handle the dynamic changes of a permutation pattern efficiently, which is described in many of the application phases [8]. The difficulty hold in the design effort to compute the routing to the permutation changes in runtime and as well as to guarantee [9] the permutated traffics for efficient support. This will become a great challenge when these permutation networks need to be implemented under very limited on-chip power and area overhead. Reviewing on-chip permutation networks (supporting either full or partial permutation) with regard to their implementation shows that most the networks employ a packet-switching mechanism to deal with the conflict of permuted data [3-6].

In this paper we present a new hardware architecture fig. 1 which is based on to improve the permutation traffic efficiency. In arbiter system we use the programmable arbiter priority (PAB) logics to produce the data under priority if at each switch no of inputs came at a time. Actually in existing system only use the fixed priority arbiter for priority based data transferring. In this paper the new proposed arbiter is with programmable logics. It provides three priority logics according to requirement priority that three priority logics are

International Journal of Advanced Research Foundation
Website: www.ijarf.com, Volume 1, Issue 2, November 2014)

F-priority (Fixed Priority), Round Robin priority (RR-priority), Dynamic Priority (D-priority). In D-priority is mixing operation of both F-priority and RR-priority to speed up the parallel operation.

So the below sections illustrates the operation of proposed architectures and switching activities well in manner. Section II describes the top of architecture and interconnections for circuit switching activity. Section III describes about the dynamic path set up scheme for programmable data transfer according to switching operations. Here it describes about the PAB (Programmable arbiter) as well. Section IV illustrates the Implementation and results. Section V illustrates the conclusion of project paper.

II. PROPOSED DESIGN

As per explanation motivated in section I is the key idea to design the proposed on chip network based on a pipelined circuit switching approach with dynamic path-setup scheme supporting runtime path arrangement. In this section II it discuss the about path set up scheme and network topology well to understand the design of OCP(on chip permutation) network. And later design of switching nodes presented well to understand.

A. On-Chip Network Topology

A family of multistage networks is clos network. It is applied to build scalable commercial multiprocessors with number (thousands) of nodes in macro systems [7], [11]. A three stage typical Clos network is defined a C (m, n, p) where the m number of inputs represents in each of first-stage switches and n is the number of second-stage switches. In order to support a parallelism degree of 16 inputs as in most practical MPSoCs [3-5], for the designed network C(4,4,4) we proposed to use as a topology (see Fig. 1).This network has a property of rearrangeable [11] that all possible permutations can realize between its input and outputs. The choice of the three-stage Close network with a modest number of middle-stage switches is to minimize implementation cost, whereas it still enables a property rearrangable for the network.

Pipelined circuit-switching scheme is designed and introduced for the proposed OCP network. This scheme has three phases: the *setup*, *transfer*, and *release* [2], [9]. A dynamic path-setup scheme *which* supporting the runtime path arrangement occurs in the phase setup. To support this circuit-switching scheme, a switch-by-switch interconnection and with its handshake signals are proposed, as Pipelined circuit switching scheme is designed and introduced for the proposed OCP network. This scheme has three phases: the *setup*, *transfer*, and *release* [2], [9]. A dynamic path-setup scheme *which* supporting the runtime path arrangement occurs in the phase setup. To support this circuit-switching scheme, a switch-by-switch interconnection and with its handshake signals are proposed, as code is also used in for both the setup and release phases. Ans=01(ack) means that the destination is ready to receive data from the source. When Ans=01 propagates back to the source, it denotes that the path is set up, then a data transfer can started immediately. An Ans=11 is reserved for end-to-end flow control when the receiving circuit is not at ready

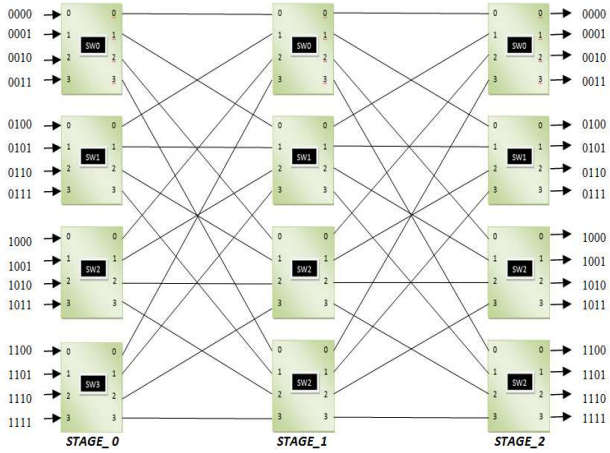

Fig. 1.Proposed OCP network of Circuit switching mechanism Architecture

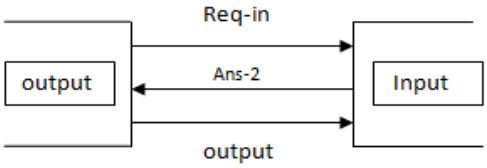

Fig. 2. The Mechanism Interface Path Diagram Of Port to Port

TABLE I SWITCH ACTIVITY

Req	***1:Setup***	***0:Idle***
1.	00:Idle	10:Back
2.	01:Ack	11:Nack

to receive the data due to it is 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 back pressure flow control of the dynamic path-setup scheme, which is discussed in the following subsection.

B. Dynamic Path Setup Scheme to Support Path Arrangement

A dynamic path set up scheme is the important point for the proposed design which to support for runtime path arrangement when the permutation is changed. In this, each and every path setup, starts from an input and find a path leading to its corresponding output, with support based on dynamic probing mechanism. The probing concept is introduced in works in [2], [9], which a probe (or setup flit) is dynamically sent under a routing algorithm in order to establish a path towards the destination. The technique Exhausted profitable backtracking (EPB) is proposed to use to route the probe in the network. An arrangement of path with *full*

International Journal of Advanced Research Foundation
Website: www.ijarf.com, Volume 1, Issue 2, November 2014)

permutation consists of sixteen path setups, where as 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 network C(m,n,p) realize all possible full permutations between its inputs and outputs? As proofed in works [11], the three-stage Clos network C(m,n,p) is rearrangeable if m greater than or equal to n. In the proposed network of C (m, n, p) m=4,n=4,p=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 always be realized in the proposed network with C (4, 4, 4) topology.

C. Switch Nodes Topology

Three kinds of switches are designed for the proposed on-chip net- work. These switches are all based on a common switch architecture shown in Fig. 3, with the only difference being in the probe routing algorithms. This common architecture has basic components: INPUT CONTROLs (ICs), OUTPUT CONTROLs (OCs), an ARBITER, and. The ARBITER has two functions: first, cross-connecting the Ans_Outs and the ICs through the Grant bus, and second, as a referee for the requests from the ICs. When an incoming probe arrives at an input, the corresponding IC examine the output status through the Status bus and through the Request bus it requests the ARBITER to grant it access to the corresponding OC. When accepting this request, the ARBITER cross-

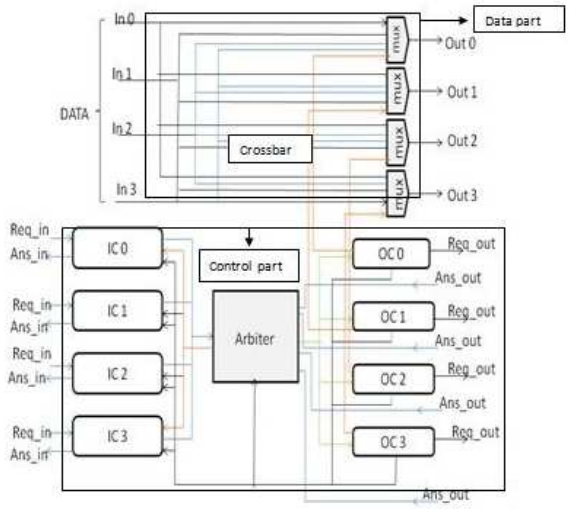

Fig. 3. Common switch architecture

With the second function, the ARBITER, based on a pre-defined priority rule, resolves Contention when several ICs request the same free output. After this resolution, only one IC is accepted, whereas the rest are answered as facing a blocked link (i.e., similar to receiving an Ans = Back).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]. In order to support the probing path setup, these 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, D2, D1, D0 (see Fig. 1 for the addressing scheme). In the first stage, the switch tries the free outputs in a non-repetitive manner (e.g., outputs 0→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 D3, D2 two most significant bits and D1, D0 the two least signification bits of the destination address are used to route the probe. Depending on the availability of the desired output or the feedback (i.e., the signal *Ans*) from the downstream switch, in a given switch the IC will change its FSM state and reply to the upstream switches accordingly. The OCs work like re-timing stages for the commands from ARBITER placed on the Control bus and control the CROSSBAR. The CROSSBAR is a 4X4 full-connecting matrix designed with output multiplexers. The ICs and the ARBITER are clocked ones with the rising and the falling edges of the clock, respectively. In this implementation, probing is processed dynamically by the switch in basis one clock cycle. As denoted in Fig. 3, the control part of switches performs the dynamic EPB-based path setup, whereas the data part provides configured paths for guaranteed data circuit-switched. This meets the target of designing the circuit-switched switches for to support EPB-based path setup in C (4,4,4) network. To validate if the designed network works as desired, a test bench is applied for to test the capability of realizing full permutation with sixteen path setups. To avoid a path setup interfering with others during the search and incurring a rearrangement of existing paths, a delay is set between the path setups launched one-by-one in a sequence in the test bench. This is to ensure that the previous path setup is completed before a new one is launched.

D. Arbiter Priorities

In this proposed network at arbiter section we proposed the three priority logics Fixed priority (F-priority),RR-priority (Round Robin Priority and D-priority (Dynamic Priority) logics which are programmable according to the priority requirement. when “00” the arbiter acts like F-priority and when “01” it acts like RR-Priority and when “11” it acts like D-priority. we know when compared to fixed priority Round Robin is efficient one .So in this paper I proposed the programmable arbiter with three priorities act .I generated output for the RR-priority in this paper.

International Journal of Advanced Research Foundation
Website: www.ijarf.com, Volume 1, Issue 2, November 2014)

III. IMPLEMENTATION ,SIMULATION AND SYNTHESIS RESULT

This proposed architecture as shown in fig. 1 is designed using verilog language and it is simulated in Modelsim software for simulation result. The below which have showed fig. 4, fig. 5, fig. 6 are the results of simulation. In paper we given "Req" enable pins at four addresses according to the above fig. 2 switch activity with dynamic path setup scheme the switches arranged the path to transfer the data from source to destination. In fig. 4 shows the Req pins enable at four addresses and data given at that point and fig. 5 results shows the switch activity enable pins like Ans_in, Ans_out and fig. 6 shows the destination address result at stage 2 from source.

The fig. 7 shows the NEXYS 2 FPGA kit of SPARTAN 3E family. On this kit implemented the above OCP architecture. The output LEDs which are blinked is the output pins of stage 2 and the switches which are enabled acts like Req's ,when the second switch from left is enabled, ie the Req pin at port of SW0 is enabled and data is transferred from source to destination depends the switching data path set up mechanism, here the output is came stage 2 switch 0 at that

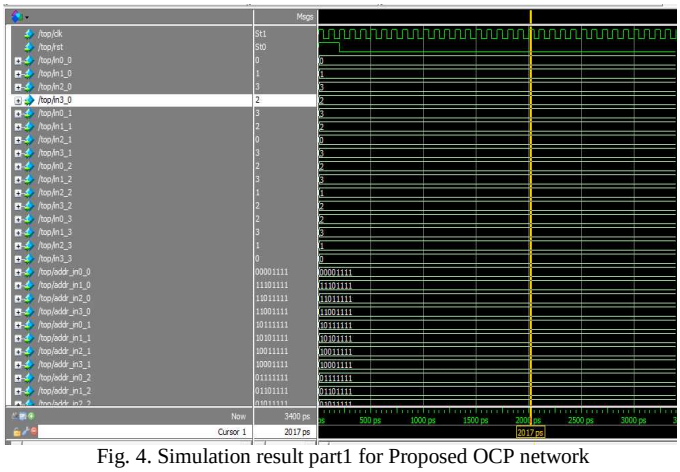

Fig. 4. Simulation result part1 for Proposed OCP network

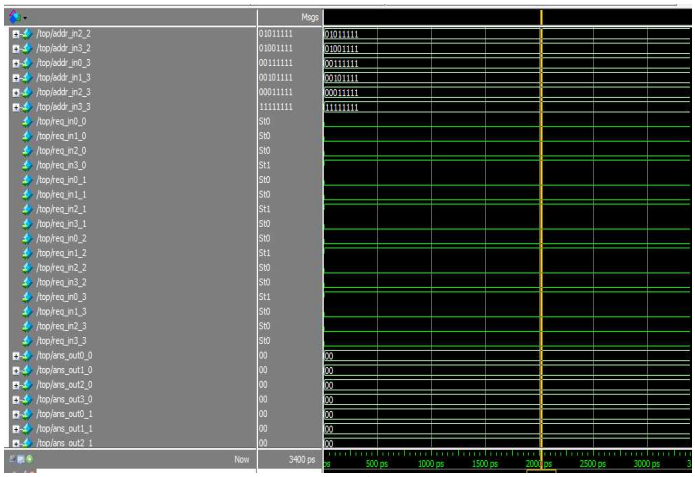

Fig. 5. Simulation Result Part2 for Proposed OCP network

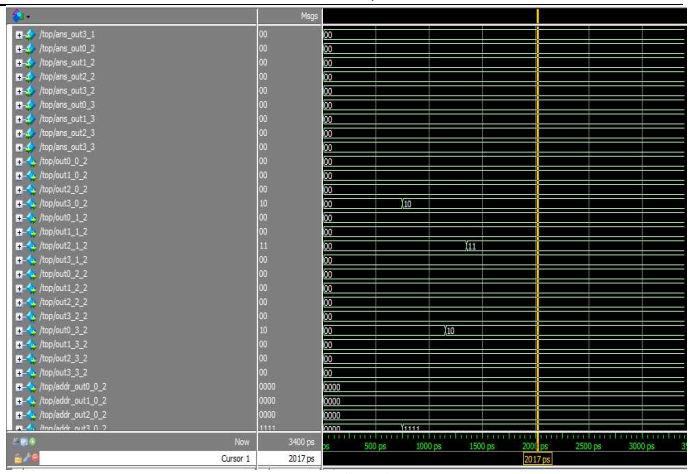

Fig. 6. Simulation Result part3for of proposed OCP Network

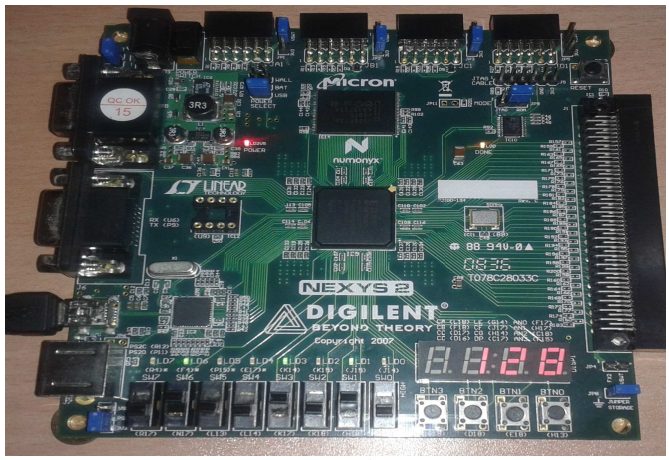

Fig. 7. FPGA Implented result of proposed OCP Network

TABLE II SYNTHESIS REPORT

Name	*Used*	*Available*	*Utilization*
No of Slices	1326	8672	15%
No of 4 input LUTs	2483	17344	14%
No of IOBs	146	250	58%
Total delay	12.624ns		

point data is "1010". And by Table II gives the synthesis report of architecture according to FPGA. In this FPGA we use device X3cs1200E and package is FG320 and Device speed is -5.

IV. CONCLUSION

This paper has presented an OCP network design supporting traffic permutations in MPSoC applications. By using a circuit switching approach combined with dynamic path-setup

scheme under a Close network topology, the proposed design offers arbitrary traffic permutation in runtime with compact implementation overhead. Design is implemented using Xilinx ISE 12.1 on FPGA Board of Spartan 3E family, NEXYS 2 kit and obtained the synthesis result regarding delay and done power analysis regarding power by that proved that efficiency is improved when compared to existing systems.

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 approachto interleaving in high throughput channel decoders," inProc.IEEEInt. Symp. Circuits Syst. (ISCAS), 2005, pp. 1766–769.

[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 topologies 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 network," 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.

Mr.K.S.V.RaghuBabu[1] received B.Tech degree in ECE from JNTUH in 2012, pursuing M.Tech (2012-2014) in the stream of VLSI at Aurora's Scientific, Technological and Research Academy, (Affiliated to JNTUH) Hyderabad.

Mrs. D.kavitha[2] received M.Tech degree in VLSI from JNTUH in 2010, B.E degree in ECE from OU in 2003, Diploma in ECE from SBTET in 1998. Presently working as Associate professor in ASTRA, Hyderabad.

Mrs.V.Rajitha Rani[3] received M.Tech degree in VLSI from JNTUH in 2010, B.E degree in ECE from JNTUH 2005,
Presently working as Sr. Assistant professor in ASTRA, Hyderabad.