IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)
e-ISSN: 2278-1676,p-ISSN: 2320-3331, Volume 9, Issue 2 Ver. V (Mar – Apr. 2014), PP 07-11
www.iosrjournals.org

Universal Functional Block Design for All Purpose Network Realization in Multi-core System

Taniza Marium, S.M Tahsin Anam, A.S.M Fahaduzzaman and Jamal Uddin Ahmed
Department of Electrical & electronic Engineering
Ahsanullah University of Science & technology
141-142, Love Road, Tejgaon Industrial Area, Dhaka, Bangladesh

Abstract*: while the last 30 years has seen the computer industry driven primarily by faster uni-processors those days have come to an end. Emerging in their place are multi processors containing multiple processor cores that are expected to exploit parallelism [3], [4]. High reliability, availability and scalability [2]needs efficient multistage switching network to ensure performance (throughput) improvement. Current advancement in IC technology has helped us to design Universal Functional\ Block for these different interconnection networks. Here in this\ paper we have designed a Universal Block for hypercube interconnection network using two different approaches indirect hypercube switching and shuffle exchange switching simultaneously.*
Index Terms*: Universal Functional Block, Indirect Hypercube, Shuffle Exchange, Omega switching.*

I. Introduction

Core-core or core-memory connects via interconnection networks. Many interconnection structures have been proposed for multicore processors which employ non-shared (static or dedicated) buses to link between neighboring cores to proceed at the maximum possible rate without interfacing from other cores. The topology or graph structure defined by the cores and their interconnection provides a useful way to characterize a multicore processor and has a major influence on its cost and performance. Common network topologies to interconnect cores include single bus, multiple busses, crossbar, tree and hypercube [1], [2], [5].Distinguishing on the basis of connection paths Static interconnection structure includes linear, mesh, ring, star, hypercube.

II. Interconnectionblockforswitching

Universal Functional Block design for interconnection occurs in a many core or multi-core system. Interconnection structures can be constructed from small interconnection networks called switching elements. Our approach is to design that Universal Functional Block where both through state and cross state switching will be operated by using this very block [2]. There is a control signal for switching elements, by setting it in several ways a large number of different interconnection patterns can be possible. These possibilities are, in general, determined by the number of stages, the fixed connection linking the stages and dynamic states of the switching elements This paper deals with the typical characteristics of some multistage switching networks, Universal Functional Block that can be used in FPGA [7] design system.

III. Hypercubenetworkusingserialorbitby BITSWITCHING

Among all the interconnection networks hypercube is very efficient for wide range of programming task. An n dimensional hypercube is shown in Fig. 1. In this system 2 cores are used each of which is connected to n neighbors. In hypercube interconnection structure number of connections is $n\log_2 n$, maximum node degree is $\log_2 n$ and maximum internodes distance is $\log_2 n$ assuming each connects n nodes [2], [5].

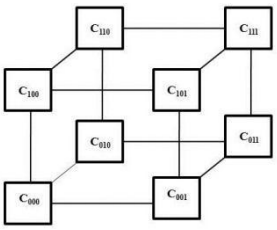

Fig. 1 Hypercube interconnection network structure.

In Fig. 2 shows the Universal Functional Block of the three stage hypercube switching network. For each stage a MUX and DEMUX is used. Particular information can be switched at different path of the network by setting

the control logic as required to meet the interconnection requests from the core and in Fig. 3 shows one of the many possible switch settings of the multi-stage switching networks of Fig. 2 [2]. Here the switching elements forming stage three are set to the cross state while the eight remaining switches are set to the through state. It can be seen that the output of C is connected to the input of C_{100} and vice versa. C_{001} is connected to the C and so on. In general switch setting of Fig. 3 simultaneously connect output of C_{ijk} to the input of C_{ijk}. On comparing this connection pattern with the state hypercube structure it is seen that interconnection network N provides all the connections linking the top four cores of the hypercube to the bottom four. Thus the parallel top-to-bottom communication allowed by the hypercube can be obtained indirectly in an eight core system that uses N as its interconnection network.

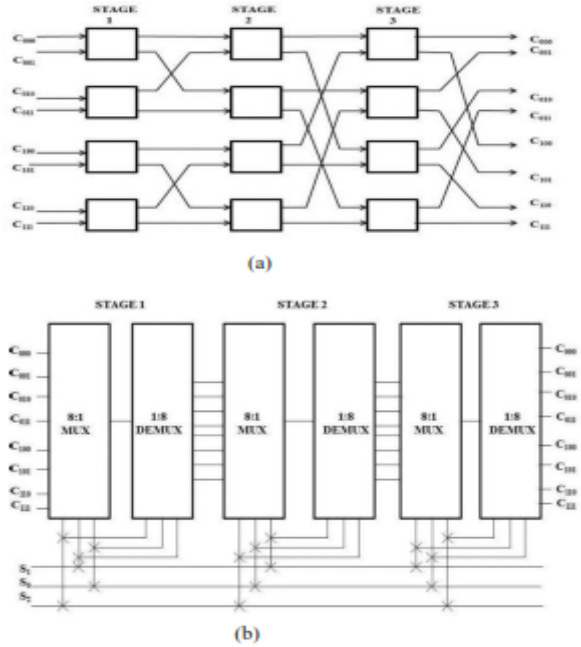

Fig. 2 Universal Functional Block for hypercube (a) network structure and (b) network design

Similarly stage one can be set to cross state while the eight remaining switches of the other two stages can be set to the through state. When stage one is set to cross state output of C_{ijk} connects with input C_{ijk} giving the connections linking the cores horizontally from left to right. The same type of parallel communication along the other dimension of the hypercube can be simulated by setting stage three at cross state .Since N provides all the interconnection pattern in this manner it is called indirect hypercube interconnection network [8],[2].

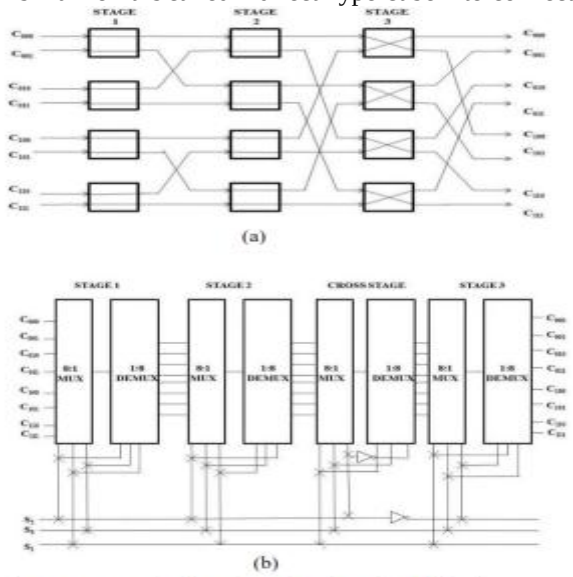

Fig. 3 One state switching of Universal Functional Block
(a) network structure and (b) network design

This indirect hypercube switching network of Fig. 2 is based on the butterfly connection depicted in Fig. 4a. It is so called because of the resemblance of the figure to a butterfly with its body in a horizontal position at the center and its wings spread out above and below. The 4 X 4 single-stage butterfly network is illustrated in Fig. 4b. Here the butterfly connection is placed before rather than after the set of N/2 switching elements. Consider an N X N multistage network with n stages 1, 2, ... , n and port addresses i = 0, 1, ..., N-1, where, i = b

$$\beta_k(b_{n-1}\dots b_{k+1}b_kb_{k-1}\dots b_1b_0)=b_{n-1}\dots b_{k+1}b_0b_{k-1}\dots {}_1b_k \qquad (1)$$

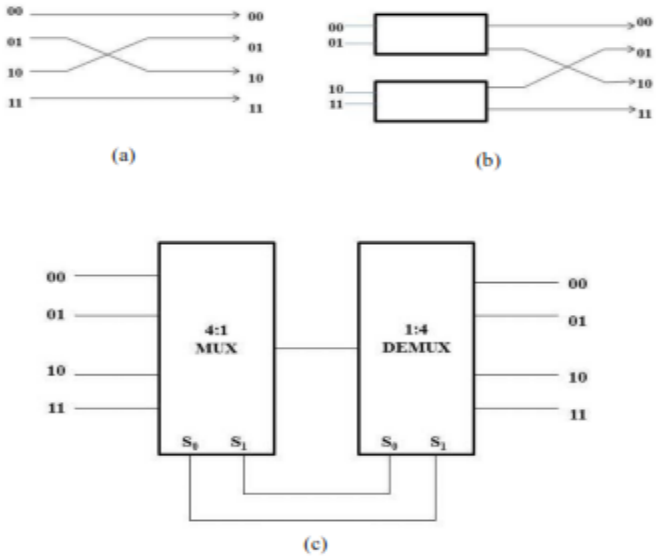

Fig. 4 (a)Butterfly connection, (b)single stage butterfly network and (c) design.

Thus interchanges bits 0 and k of the source address to obtain the destination address. For example, when k = 1 and
N = 4, we obtain

$$\beta_1(00) = 00$$
$$\beta_1(01) = 10$$
$$\beta_1(10) = 01$$
$$\beta_1(11) = 11$$

Corresponding to the interconnection pattern on Fig. 4a It iseasily seen that the Universal Functional Block of Fig. 2 , thewiring patterns following stages 1 and 2 are defined by ß₁, andß, respectively, with N = 8. Inverse shuffle, which isdefined by the following inverse s h u f fl e functions forms t hest age 3 connect i on pattern of Fig. 2 [2],
[9].

$$\sigma^{-1}(i) = b_0b_{n-1}b_{n-2}b_{n-3}\dots\dots b_1 \qquad (2)$$

IV. Shuffle Exchange (Also Known As OMEGA SWITCHING)

In Fig. 5a shows the shuffle connection. It can be defined
by the following mapping function [2], [5], [9]:

$$\sigma(i) = 2i + [2i/N] \quad (\text{modulo } N) \qquad (3)$$

The name shuffle comes from the fact that the destination addresses can be mapped into the source addresses by inter leaving the first half of the port address sequence with the second half in the manner of a perfectly shuffled deck of cards. Let each address i be represented by the corresponding n-bit binary number b. So equation 3 can be also represented as [2]:

$$\sigma(i) = b_{n-2}\, b_{n-3} \ldots b_0 b_{n-1} \qquad (4)$$

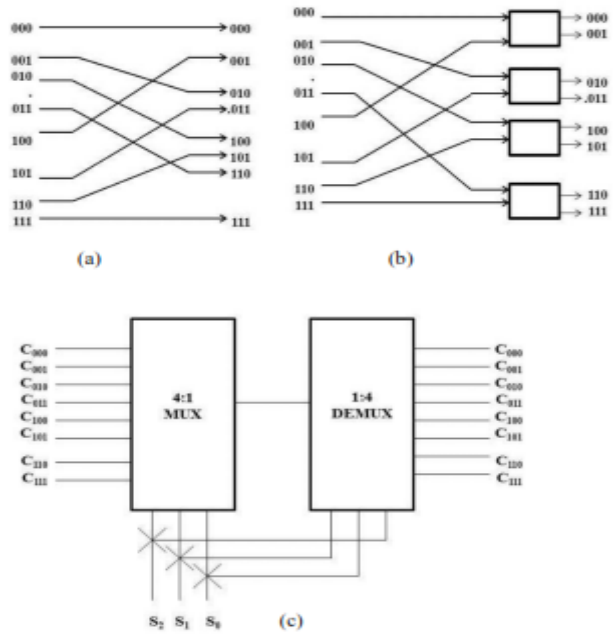

Fig. 5 (a) Shuffle connection, (b) single stage shuffle exchange network and (c) design.

Equation 4 indicates that the shuffle function corresponds to using a one bit left rotation of the source address bits to determine the destination address. By following a shuffle connection with N/2 switching elements, each of which can exchange (cross) a pair of buses, we obtain the single-stage Shuffle-exchange network, which is illustrated in Fig. 5b for the case N=8. A network constructed from n=log N cascaded shuffle-exchange stages is called an omega network [2], [9]. It is depicted in Fig. 6.It is a full access network. In Fig. 7 shows one of the states of omega switching of Fig. 6.

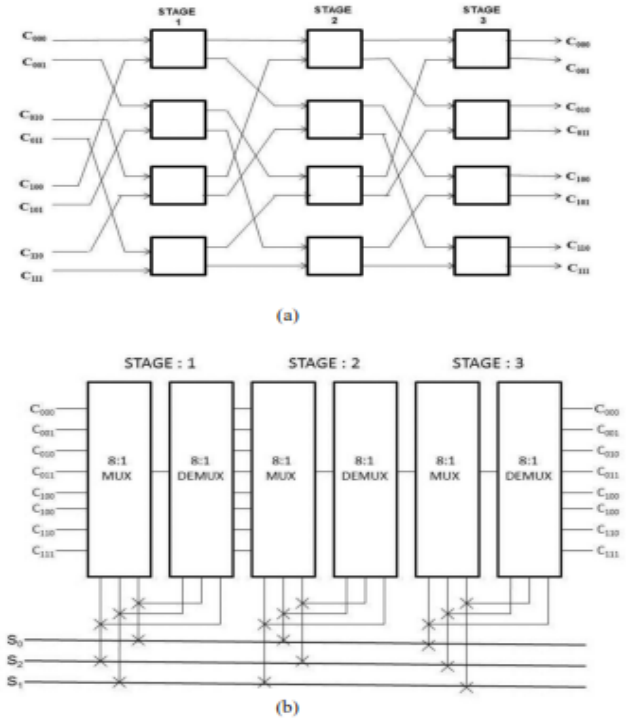

Fig. 6 (a) Omega switching network (b) design

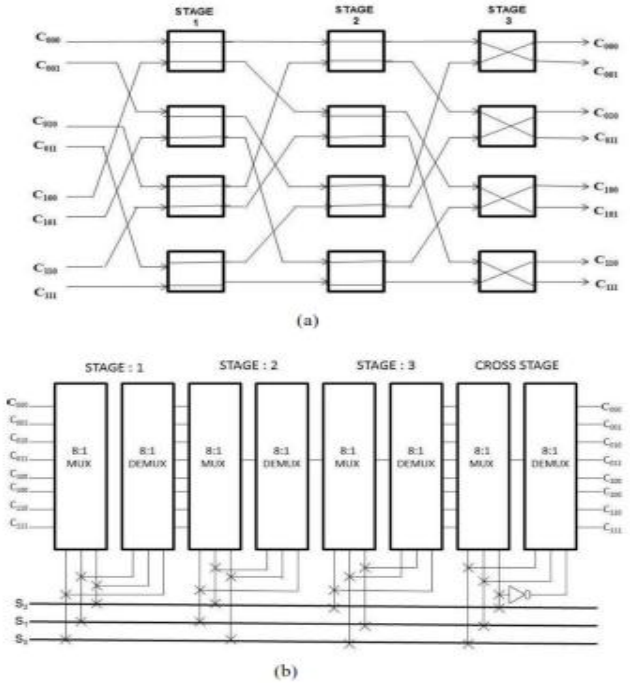

Fig. 7 (a) One state of Omega switching and (b) design

Indirect hypercube and shuffle-exchange networks have very similar properties. Suppose that the directions of all the arrows in an N X N shuffle-exchange net work are reversed, implying that the shuffle connection s in each stage replaced by σ^{-1}. The resulting N X N inverse omega network and the N×N indirect hypercube network are essentially the same multistage switching network drawn in different ways.

Designs for all these various networks need MUX and DEMUX with dynamic connection pattern. Realization of such network is possible using FPGA [7] IC programming them according to required network pattern. So same IC could be used for several types of network system.

IV. Conclusion

In this paper we have proposed the very general class of block named Universal Functional Block which is used for designing the hypercube network. Same type of functional blocks can also be designed for other networks using MUX and DEMUX. Here we did not include the circuit analysis, time analysis and performance analysis of these interconnection network designs. Most of the network design presented here are based on three bits information. In reality we can use this as a functional block and design for higher bits information. The price paid for this interconnection flexibility is the communication delay caused by the switches and some time complex algorithms needed to control their states. These allow the multi –core processors to have fast access to their local memory modules by effectively bypassing the interconnection network. Having all cores be the same makes production easier and keeps its complexity to a minimum.

References

[1] MULTIPROCESSORS INTERCONNECTION NETWORKS.Available: http://lyle.smu.edu/~rewini/slides/adv-arch bk/chapter02.pdf
[2] John P.Hayes, *Computer architecture and organization (second edition)*
[3] Stephen W. Keckler, KunleOlukotun, H. Peter Hofste, *Multi-core processors and systems.*
[4] Cameron Hughes Tracey Hughes, *Professional Multicore Programming Design and Implementation.*
[5] Interconnection Network, Available: www.ece.eng.wayne.edu/~czxu/ece7660_f05/network.pdf
[6] Laxmi N. Bhuyan, Member, IEEE, And Dharma P. Agarwal, Senior Member, IEEE, *Generalized Hypercube and Hyper bus Structures for a Computer Network.*
[7] Sung-Mo Kang and Yusuf Leblebici, *CMOS Digital Integrated Circuits(third edition).*
[8] Andy Pimentel, *Introduction to Parallel Architecture.*
[9] Nur Arzilawati MdYunus and Mohamed Othman, *Shuffle Exchange Network in Multistage Interconnection Network: A Review and Challenges.International Journal of Computer and Electrical Engineering, Vol. 3, No. 5, October 2011*