Journal of Engineering Sciences
Vol 11, Issue 7, July /2020
ISSN NO:0377-9254

Standard basic Code Based encoding/Decoding with AND Operation for Network on Chip Communication

[1] KRISHNA BANAVATHU, [2] ALIKANI VIJAYA DURGA
[1]Assistant Professor, Dept. of ECE, University College of Engineering Adikavi Nannaya University, Rajamahendravaram, AP.
[2] Assistant Professor, Dept. of ECE, University College of Engineering Adikavi Nannaya University , Rajamahendravaram, AP.

Abstract: As a high performance on-chip communication method the code division multiple access (CDMA) technique has recently been applied to networks on chip (NOCs). We propose a new standard-basis based encoding/decoding method with test circuit to leverage the performance and cost of CDMA NOCs in area, power assumption, and network throughput. on-line transparent test technique for detection of latent hard faults which develop in first input first output buffers of routers during field operation of NoC. The technique involves repeating tests periodically to prevent accumulation of faults. In the transmitter module, source data from different senders are separately encoded with an orthogonal code of a standard basis and these coded data are mixed together by an XOR operation. Then the sums of data can be transmitted to their destinations through the on chip communication infrastructure. In the receiver module, a sequence of chips is retrieved by taking an AND operation between the sums of data and the corresponding orthogonal code. After a simple accumulation of these chips, original data can be reconstructed. We implement our encoding/decoding method and apply it to a CDMA NOC with a star topology. Compared with the state-of-the-art Walsh-code-based (WB) encoding/decoding technique, our method achieves power saving and area saving together with decrease of encoding/decoding latency. Moreover, the CDMA NOC with different sizes applying our encoding/decoding method gains power saving, area saving, and maximal throughput improvement respectively, than the WB CDMA NOC. The proposed architecture of this paper analysis the logic size and area using Xilinx 14.6.

1. INTRODUCTION

Over the last decade, network-on-chip (NoC) has emerged as a better communication infrastructure compared with bus-based communication network for complex chip designs overcoming the difficulties related to bandwidth, signal integrity, and power dissipation [1]. However, like all other systems-on-a-chip (SoCs), NoC-based SoCs must also be tested for defects. Testing the elements of the NoC infrastructure involves testing routers and inter router links. Significant amount of area of the NoC data transport medium is occupied by routers, which is predominantly occupied by FIFO buffers and routing logic. Accordingly, the probabilities of run-time faults or defects occurring in buffers and logic are significantly higher compared with the other components of the CDMA based NoC. Thus, test process for the NoC infrastructure must begin with test of buffers and routing logic of the routers. In addition, the test must be performed periodically to ensure that no fault gets accumulated.

With the rapid growth of the computational complexity more and more processing elements (PEs) are integrated onto a single chip and network on chip (NoC) has been proposed to address the versatility throughput and unwavering quality issues of on-chip communication. However ordinary packet-switched NoCs experience the ill effects of nondeterministic transmission idleness and constrained open doors for parallel information exchange, since different streams can't overcome a connection in the meantime. To determine these issues the CDMA system as a compelling technique for actualizing elite on-chip communication was connected to NoCs. The beforehand proposed CDMA NoCs depend on a computerized encoding and translating technique requiring that the spreading codes have both the orthogonal and adjust properties. To this end the Walsh code is ordinarily utilized. However the Walsh-code-based (WB) encoding and translating strategy has inalienable deficiencies which are given as takes after.

1**)** ***Design Complexity:*** In the encoding strategy, a math expansion logic unit, whose logic overhead increments with the quantity of senders, is utilized to combine coded information. In the interpreting strategy a key demux accumulation contrast unit is utilized with recover the source information from blended information chips (in this short each piece of a spreading code is known as a chip and hence the encoded information is called information chips). This unit is in any case, zone expending.

2) ***Low Code Utilization:*** In a S-chip Walsh code set S must be equivalent to 2N, where N is a characteristic number, and at most S − 1sequences can be utilized to encode the first information. This outcomes in a misuse of arrangements in the code set. For instance a 16-hub organize needs a 32-chip Walsh code set on the grounds that a 16-chip Walsh code set can just give 15 arrangements to information encoding and it in this manner can't fulfill the necessity of 16 successions one for every hub.To address the previously mentioned shortcomings, we propose another standard basic Based (SB) encoding/translating technique, which beats the WB encoding/deciphering strategy. The SB encoding/interpreting strategy can be connected to any CDMA NoCs to enhance their execution. The CDMA technique is becoming extremely popular in interconnecting mechanism of IP core for its

efficient utilization of the data transfer among the IPs, to achieve higher throughput and less latency currently silicon chips that contain more and more number of transistors with 45nm feature size are available in the market, according to the report, the International Technology Road map for Semiconductors (IRTS), a single semiconductor chip will contain multi-billion transistors with a feature size around 22nm and clock frequency of nearly 35 GHz by the year of 2016.

This growing manufacture capacity and demanding applications continuously increase the complexity of an System-on-Chip (SoC) to a higher degree in terms of number of system components and functionalities. Currently used bus architectures like Core connect for data transfer in an on-chip system has several disadvantages, bus arbitration and bandwidth limitations. In order to overcome the disadvantages of bus structure, the concept of Network-on-Chip (NoC) has been proposed as a solution to interconnect variable IP Core in system architecture.

The early design of NoC has been emerged from SoC where the router is considered as a software running on to perform routing. The space and cost of SoC systems have motivated the researchers to find an alternate medium for performing the specified task with least cost and maintenance. The NoC is a combination of various components like buffer, processing unit and scheduling program are encoded with the design. The NoC systems replaced SoC systems based on the cost and maintenance. The purpose of NoC routers is to perform routing of the packet between various nodes of the network which are raised from various processes.

General working principle of CDMA

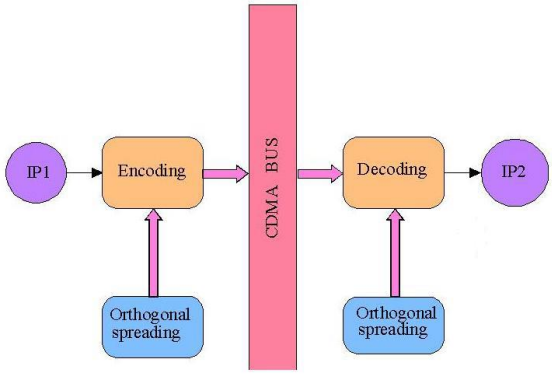

Fig: 1The general working of CDMA technology

The general working of CDMA technology has two stages namely encoding and decoding as shown in Fig 1. The source node encodes the message utilizing the key selected, which is performed based on the orthogonal spreading codes. The orthogonal spreading codes maintain the uniqueness of the signature which will be distinct to each process and could be decoded by the pair of process.The processes use the code to encode the message and each process uses different code which is visible for the subsequent pair of process. The messages or packets which are transmitted in the same channel will not feed the process and meshes up. At the sending end, the information from various senders are encoded utilizing an arrangement of orthogonal spreading codes. The encoded information from various senders are included for transmission without meddling with with each other as a result of the orthogonal property of spreading codes.

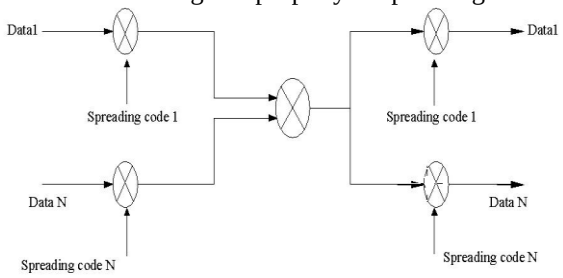

Fig: 2 Working principle of CDMA

In light of the orthogonal property, at the less than desirable end, the information can be decoded from the gotten whole flags by duplicating the got signals with the spreading code that are utilized for encoding. In the encoding plan, information from various senders are sustained into the encoder a little bit at a time. Every datum bit will be spread in to S bits by XOR logic operations with an exceptional S-bit spreading code as showed in Figure 2. Each piece of the S-bit encoded information produced by XOR operations is known as an information chip. At that point the information chips which originates from various Senders are included mathematically as per their bit positions in the S-bit successions. In particular, all the main information chips from various senders are included and all the second information chips from various senders are included together thus.

Therefore after the add operations, we will get S sum values of S bit encoded data. Then the binary equivalents of each sum value will be transferred to the receiving end. The scheme of applying the CDMA technique into on-chip communications has already been - proposed in several works. These works concern analog circuits to implement the CDMA technique, the representation of encoded data is in the form of continuous voltage or capacitance value of the circuits.

However the data transfers through the analog circuits are challenged - in terms of coupling noise, clock skew problems and the variation of capacitance caused by manufacturing processes. Hence, a method of applying digital CDMA technique is proposed in this work. Figure 2.3 shows the general working principle of CDMA technique where the input data is encoded with the spreading code selected and forwarded to the receiving side.

The received data will be decoded with the same on the other side.

2. IMPLEMENTATION OF THE TEST ON FIFO BUFFERS OF NOC ROUTERS

In this section, we present the technique used for implementing the proposed transparent SOA-MATS++ test on a mesh-type NoC. Data packets are divided into flow control units (*flits*) and are transmitted in pipeline fashion [1]. The flit movement in a mesh-type NoC infrastructure considered for this work is assumed to require buffering only at the input channels of routers. Thus, for a data traffic movement from one core to another, the online test is performed only on the input channel FIFO buffers, which lie along the path. The buffers operate in two modes, the *normal* mode and the *test* mode. The normal mode and test mode of operation of a FIFO buffer are synchronized with two different clocks. The clock used for test purpose (referred as *test_clk* in this brief) is a faster clock compared with the clock required for normal mode (*router clock*).

The FIFO buffers are allowed to be operative in normal mode for sufficient amount of time before initiating their test process. This delay in test initiation provides sufficient time for run-time intermittent faults developed in FIFO buffers to transform into permanent faults. The test process of a targeted FIFO buffer is initiated by a counter, which switches the FIFO buffer from normal mode to test mode.

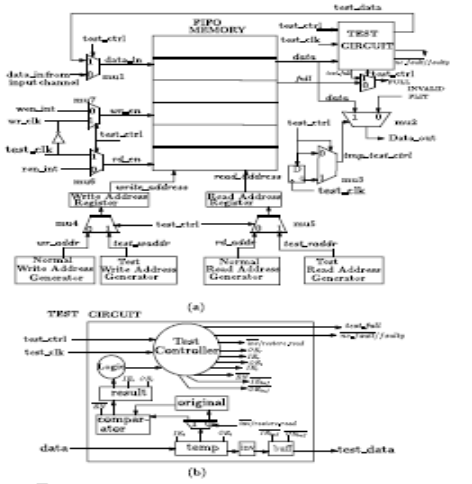

Fig: 3. (a) Hardware implementation of the test process for the FIFO buffers. (b) Implementation of test circuit.

The switching of FIFO buffers from normal mode to test mode occurs after a certain period of time without caring about the present state of the FIFO buffer. It may be argued that at the instant of switching, the buffer may not be full, and as a result not all locations would be tested during the test cycle. However, test initiation after the buffer gets full would cause the following problems. First, wait For the buffer to get full would unnecessarily delay the test initiation process and would allow faults to get accumulated. Second, test of the entire buffer would prolong the test time and would negatively affect the normal mode of operation.

A test burst involves series of test read and write cycles. It requires three read and two write cycles, or in other words three cycles of the faster test clock to perform a transparent SOA-MATS++ test on a single location of a FIFO buffer. It may be argued that during a test burst, not all FIFO buffer locations are tested or a test of a location can get interrupted. These two problems can be avoided by periodically testing the FIFO buffers. Periodic testing of a FIFO buffer allows test of a different set of locations of the FIFO buffer in each test burst. Every time the buffer is switched to test mode, the

normal process gets interrupted. The FIFO memory location currently addressed in normal mode, at the instant of switching, becomes the target location for test. Since normal operation is interrupted at different instants in different test bursts, the locations tested in each burst would be different. Thus, repeating the test bursts for a number times on a FIFO buffer would cover the test of each location as the number of locations in a FIFO buffer is few. Moreover, periodic testing prevents accumulation of fault in the buffer.

A. Test Architecture

The FIFO buffer present in each input channel of an NoC router consists of a SRAM-based FIFO memory of certain depth. During normal operation, data flits arrive through a *data_in* line of the buffer and are subsequently stored in different locations of the FIFO memory. On request by the neighboring router, the data flits stored are passed on to the output port through the *data_out* line. Fig. 3(a) shows the FIFO memory with *data_in* and *data_out* line. To perform the transparent SOA-MATS++ test on the FIFO buffer, we added a test circuit, few multiplexers and logic gates to the existing hardware, as shown in Fig. 3(a). The read and write operations on the FIFO buffer are controlled by the read enable and write enable lines, respectively. The multiplexers *mu*6 and *mu*7 select the read and write enable during the normal and test process. During normal operation when the *test_ctrl* is asserted low, the internal write and read enable lines, *wen_int* and *ren_int*, synchronized with the router clock, provide the write and the read enable, respectively. However, during test process, the write enable and read enable are synchronized with the test clock. As mentioned earlier, the read and write operations during test are performed at alternate edges of a test clock. The read operations are synchronized with the positive edges, while the *write_clk* is obtained by inverting the test clock. In test mode (*test_ctrl* high), the test read and write addresses are generated by test address generators implemented using gray code Counters similar to the normal address generation. Muxes *m*4 and *m*5 are

used to select between normal addresses and test addresses.

Consider the situation when the FIFO buffer is in normal mode with flits being transferred from the memory to the *data_out* line. After a few normal cycles, the *test_ctrl* is asserted high, switching the buffer to test mode. As long as the buffer is in test mode, no external data is allowed to be written to the buffer, or in other words, the buffer is locked for the test period. As a result, the input data line for the FIFO memory is switched from the external *data_in* line to *test_data* line available from the test circuit. At the switching instant, the flit which was in the process of being transferred to the *data_out* line is simultaneously read into the *Test Circuit*. However, a one clock cycle delay is created for the flit to move to the *data_out* line. This delay ensures that the flit is not lost during the switching instant and is properly received by the router, which requests for it. The single cycle delay in the path of the traveling flit is created by the D-type flip-flop and the multiplexer *m*3, as shown in Fig. 3(a). The flit, which is read in the test circuit, is stored in a temporary register *temp* and the test process begins with this flit. To avoid packet loss during testing, the *FULL* signal of the FIFO is asserted *high* so that neighboring routers can be prevented from transferring packets to the corresponding router. However, applying such technique increases the network latency

3. Simulation Results

In this section, we present the technique used for implementing the proposed transparent SOA-MATS++ test on a mesh-type NoC. Data packets are divided into flow control units (*flits*) and are transmitted in pipeline manner. The flit movement in a mesh-type NoC infrastructure considered for this work is assumed to require buffering only at the input channels of routers. Thus, for a data traffic movement from one core to another, the online test is performed only on the input channel FIFO buffers, which lie along the path. The buffers operate in two modes, the *normal* mode and the *test* mode. The FIFO buffers are allowed to be operative in normal mode for sufficient amount of time before initiating their test process. This delay in test initiation provides sufficient time for run-time intermit-tent faults developed in FIFO buffers to transform into permanent faults. The test process of a targeted FIFO buffer is initiated by a counter, which switches the FIFO buffer from normal mode to test mode. The switching of FIFO buffers from normal mode to test mode occurs after a certain period of time without caring about the present state of the FIFO buffer. It may be argued that at the instant of switching, the buffer may not be full, and as a result not all locations would be tested during the test cycle. . However, test initiation after the buffer gets full would cause the following problems. First, wait for the buffer to get full would unnecessarily delay the test initiation process and would allow faults to get accumulated. Second, test of the entire buffer would prolong the test time and would negatively affect the normal mode of operation.

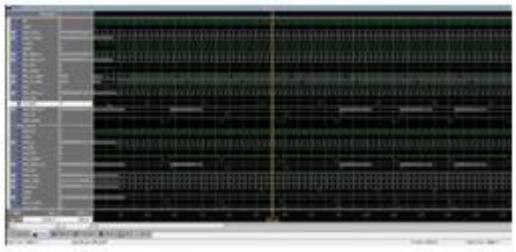

Fig.4. Simulation Results

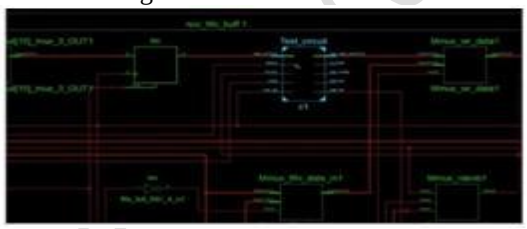

Fig.5. Test Diagram for FIFO Buffers of NOC Routers

Project File:	Online_NOC.xise	**Parser Errors:**	No Errors
Module Name:	Top_module	**Implementation State:**	Placed and Routed
Target Device:	xc3s100e-5vq100	• **Errors:**	No Errors
Product Version:	ISE 14.6	• **Warnings:**	21 Warnings (1 new)
Design Goal:	Balanced	• **Routing Results:**	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	• **Timing Constraints:**	All Constraints Met
Environment:	System Settings	• **Final Timing Score:**	0 (Timing Report)

Device Utilization Summary

Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Latches	24	1,920	1%	
Number of 4 input LUTs	38	1,920	1%	
Number of occupied Slices	26	960	2%	
Number of Slices containing only related logic	26	26	100%	
Number of Slices containing unrelated logic	0	26	0%	
Total Number of 4 input LUTs	38	1,920	1%	
Number of bonded IOBs	25	66	37%	
IOB Latches	1			
Number of BUFGMUXs	3	24	12%	
Average Fanout of Non-Clock Nets	1.85			

Fig: 6 Device utilization in Xilinx

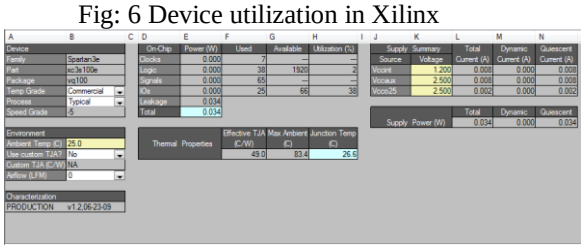

Fig: 7 power report of NOC

Power value =Maximum power achieves from the both decoder and encoder of existing method - Maximum power achieves from the both decoder and encoder of proposed method /Maximum power achieved.

CONCLUSION

In this paper Standard Basis Code Multiplexer using test circuit operation for an on chip interconnection network has been presented. Standard basis codes are used to modulate the packet data to handle a large number of parallel data transfer. Compare to the existing standard basis code method, the proposed standard basis code of CDMA approach provides reduction in delay, power, and area and achieves maximum throughput. A Standard Basis Code Multiplexer using test circuit operation for an on chip achieves very less delay than the

existing method. In proposed method low number of gates is used than the existing method. So, it occupies less area than existing one. Normally multiplexer is used in many areas for data transferring. Multiplexer sends digital or analog signals at higher speed on a single line in one shared device.

FUTURE SCOPE

Future work includes Design of Low Power & Reliable Networks on Chip through Joint Crosstalk Avoidance and Multiple Error Correction Coding.

REFFERENCES

[1] D. Sigüenza-Tortosa, T. Ahonen, and J. Nurmi, "Issues in the development of a practical NoC: The Proteo concept," *Integr., VLSI J.*, vol. 38, no. 1, pp. 95–105, 2004.

[2] A. J. Viterbi, *CDMA: Principles of Spread Spectrum Communication*. Reading, MA, USA: Addison-Wesley, 1995.

[3] S. Shimizu, T. Matsuoka, and K. Taniguchi, "Parallel bus systems using code-division multiple access technique," in *Proc. Int. Symp. Circuits Syst.*, May 2003, pp. II-240–II-243.

[4] D. Kim, M. Kim, and G. E. Sobelman, "CDMA-based network-on-chip architecture," in *Proc. IEEE Asia-Pacific Conf. Circuits Syst.*, Dec. 2004, pp. 137–140.

[5] X. Wang and J. Nurmi, "An on-chip CDMA communication network," in *Proc. Int. Symp. Syst.-Chip*, Nov. 2005, pp. 155–160.

[6] E. H. Dinan and B. Jabbari, "Spreading codes for direct sequence CDMA and wideband CDMA cellular networks," *IEEE Commun. Mag.*, vol. 36, no. 9, pp. 48–54, Sep. 1998.

[7] M. Kim, D. Kim, and G. E. Sobelman, "MPEG-4 performance analysis for a CDMA network-on-chip," in *Proc. Int. Conf. Commun., Circuits, Syst.*, May 2005, pp. 493–496.

[8] X. Wang, T. Ahonen, and J. Nurmi, "Applying CDMA technique to network-on-chip," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 15, no. 10, pp. 1091–1100, Oct. 2007.

[9] W. Lee and G. E. Sobelman, "Mesh-star hybrid NoC architecture with CDMA switch," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2009, pp. 1349–1352.

[10] M. Kim, D. Kim, and G. E. Sobelman, "Adaptive scheduling for CDMA-based networks-on-chip," in *Proc. 3rd Int. IEEE-NEWCAS Conf.*, Jun. 2005, pp. 357–360.

[11] W. Lee and G. E. Sobelman, "Semi-distributed scheduling for flexible codeword assignment in a CDMA network-on-chip," in *Proc. IEEE 8th Int. Conf. ASIC*, Oct. 2009, pp. 431–434.

[12] S. Poddar, P. Ghosal, P. Mukherjee, S. Samui, and H. Rahaman, "Design of an NoC with on-chip photonic interconnects using adaptive CDMA links," in *Proc. IEEE Int. Conf. SOC*, Sep. 2012, pp. 352–357.

KRISHNA BANAVATHU is a Assistant professor in University College of Adikavi Nannaya University. He Received B. Tech in Electronics and Communication Engineering from RVR &JC College of Engineering, Guntur in 2007. He Received his M. Tech in 2013 From Sasi Institute of Engineering And Technology, JNTUK. He Currently Research in Image Processing and Very large Scale integrated Circuits.

ALIKANI VIJAYA DURGA is a Assistant professor in University College of Adikavi Nannaya University. He Received B. Tech in Electronics and Communication Engineering from Godavari And Technology Affiliated to JNTUK in 2010. He Received his M. Tech in 2014 From Chaitanya Institute of Science And Technology Affiliated to JNTUK. He Currently Research in Image Processing and Very large Scale integrated Circuits.