Network on Chip with CDMA Technique

V Ravichandran* and G K D Prasanna Venkatesan

*1Department of Electronics and Communication, MIET Engineering College, Tamil Nadu-620 007, India
2Department of Electronics and Communication, PGP College of Engineering &Technology, Tamil Nadu-637 207, India

Received 20 July 2012; revised 20 August 2013; accepted 20 December 2013

NoC (network On Chip) is an efficient approach to design the communication subsystem between IP Cores in SoC (System on Chip). In this paper a communication infrastructure design using CDMA (Code division multiple access) based shared bus architecture for core-to-core communication in NoC is presented. CDMA has been proposed as an alternative way for interconnect of IP cores in a SoC design, or as a solution for interconnecting modules within a system realized in several PCBs. Compared to a conventional TDMA-based bus this paper present how a multiprocessor system can benefit from the use of concurrent data transfers. The simulation of the system using VHDL code is carried out. The performance of the system and is evaluated.

Keywords: CDMA bus, Decoding, Encoding, NOC, Walsh code

Introduction

A SoC consists of processing elements, I/O devices, Storage elements and integration structure linking all of them together. Processing elements could be processors that run embedded software or functional-specific hardware accelerators. As the number of processing elements increases, the interconnection plays an increasingly major role in system on-chip (SoC) design. The type of interconnection used for a specific application will heavily affect the performance and power consumption of the system. A variety of interconnection schemes is currently in use, including crossbars, rings, buses, and networks-on-chip. The shared-buses are among the most widely used communication architectures in systems-on-chip. The main advantages of shared-bus architectures include simple topology, low cost, and extensibility. Several companies have developed their own on-chip bus architectures, such as Core-Connect, AMBA, and Silicon Micro-Networks.

The main problem of shared-bus communications is that the performance of a bus decreases significantly when the bus size increases. This is the so-called scalability problem. Because a bus can be used by only one module at the time, the available bandwidth of one module decreases significantly as the bus size increases. The bandwidth can be improved by hierarchical bus architecture where multiple buses are connected with each other through bridges. However, hierarchical bus architectures may suffer from long communication latency for inter-bus communications.

Network-On Chip

To overcome such problems, Research groups have started to investigate systematic approaches to the design of the communication part of SoCs. It turned out that the problem has to be addressed at all levels from the physical to the architectural to the operating system and application level.

Over the past few years, the idea of using networks-on-chip (NoCs) as viable on-chip communication fabrics for future multiprocessor systems-on-chips (MPSOcs) has been gaining momentum. NoCs are an attempt to scale down the concepts of large scale networks, and apply them to the embedded system-on-chip (SoC) domain. Unlike traditional bus-based on-chip communication architectures, NoCs use packets to route data from the source to the destination component, via a network fabric that consists of switches (routers) and interconnection links (wires)

Hence, the term Network on Chip (NoC) is today used mostly in a very broad meaning, encompassing
the hardware communication infra-structure, the middleware and operating system communication services and a design methodology and tools to map applications onto a NoC. All this together can be called a NoC platform. To overcome the above mentioned limitations, the attention has shifted towards the other popular interconnection architecture: network-on-chip (NOC). This communication scheme assumes that the data is packetized and then transmitted within a chip through a network.

Experimental Section
On Chip Interconnect Mechanism
In general, on-chip communication architectures can be categorized into three main classes namely Bus, Point to Point interconnect and Network-on-Chip. However, there is no standard solution how to establish fast, flexible, efficient, and easy to design communication network to connect a large number of IP cores that have heterogeneous requirements.

TDMA based Interconnection
Most of interconnect networks in the SoC rely on parallel system bus, which apply Time division multiple access (TDMA). The popularity and wide acceptance of the shared bus architecture is perhaps due to the fact that it is easy to adopt, well known among the computer industry, and also relatively inexpensive to implement. In those systems, the bus masters perform read and write operations with slave memory or I/O modules. However, bus-based architectures cannot scale up with an increasing number of components.

CDMA Based Interconnection
CDMA is a spread-spectrum technique that allows simultaneous use of the communication medium by multiple information links. It relies on the principle of codeword orthogonality, that avoids cross-correlation of code-words and allows perfect separation of information modulated with different code-words. The Code Division Multiple Access, (CDMA), is used to implement the memory/peripheral shared bus of multi-core systems consisting of multiple processing cores. In this method only bus lines that carry address and data signals are CDMA coded.

Motivation for using CDMA Technique
During realization of high speed buses we meet with numerous problems. These problems are typical for realization of a interconnection. To solve this problem various techniques are used. In most cases, these techniques include implementation of additional hardware. An alternative solution to increase bus throughput consists of increasing bus data transfer lines.

As a consequence, by using this approach, the number of lines, system complexity, occupied PCB area, and PCB tracing increase. In all cases, the buses with corresponding interfaces become very complex system. As a number of bus lines becomes higher the cost of these systems increases. Bearing this in mind, a bandwidth improvement achieved by increasing the number of bus lines, for most design solutions, is not a rational economical solution.

Compared to a conventional TDMA-based bus, a CDMA-based bus has better features concerning channel's isolation and channel's continuity in time domain since channels are divided by the spreading codes. CDMA technology relies on the principle of code word orthogonally, such that it enables efficient separation of information. In our proposal, only bus lines that carry address and data signals are CDMA coded.

Principle of the CDMA Technique
CDMA Principle
The principle of the CDMA technique is illustrated in (Fig. 1). At the sending end, the data from different senders are encoded using a set of orthogonal spreading codes. The encoded data from different senders are added together for transmission without interfering with each other because of the orthogonal property of spreading codes. Because of the orthogonal property, at the receiving end, the data can be decoded from the received sum signals by multiplying the received signals with the spreading code used for encoding. In the encoding scheme data from different senders fed into the encoder bit by bit. Each data bit will be spread into S bits by XOR logic operations with a unique S-bit spreading code as illustrated in (Fig. 2). Each bit of the S-bit encoded data generated by XOR operations is called a data chip. Then, the data chips which come from different senders are added together arithmetically according to their bit positions in the S-bit sequences. Namely, all the first data chips from different senders are added together and all the second data chips from different senders are added together, and so on.

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. In this case, two binary bits are enough to represent the three possible different decimal sum values, “0,” “1,” and “2.” For example, if a decimal sum value “2” needs to be transferred, we need to transfer two binary digits “10”.

The CDMA encoding scheme is explained with example. Let us consider two data bits (1, 0) from Sender1; Sender2 is XOR with two different 8-bit spreading codes 10101010, 11001100 respectively. Then the encoded data bit (Data chips) for Sender1 & Sender2 are added arithmetically. The sum value (12011201) is converted to 2-bit binary value as 0110000101100001. The decoding scheme can be explained as follows. If the original data bit to be transferred is “1,” after the XOR operations in the encoding scheme illustrated in (Fig. 2), it can only contribute nonzero value to the sums of data chips when a bit of spreading code is “0.” Similarly, the 0-value original data bit can only contribute nonzero value to the sum of data chips when a bit of spreading code is “1.” Therefore, after accumulating the sum values according to the bit values of the spreading code, either the positive part or negative part is larger than the other if the spreading codes are orthogonal and balance. Hence, the original data bit can be decoded by comparing the values between the two accumulators. Namely, if the value of the positive accumulator is larger than the value in the negative accumulator, the original data bit is “1”; otherwise, the original data bit is “0”.

**Spreading Code Selection**

The proposed decoding scheme requires the spreading codes used in the CDMA NoC to have both the orthogonal and balance properties. Several types of spreading codes have been proposed for CDMA communication, such as Walsh code, M-sequence, Gold sequence, and Kasami sequence, etc. However, only Walsh code has the required orthogonal and balance properties. Therefore, Walsh code family is chosen as the spreading code library for the CDMA NoC.

**CDMA bus for on chip Communication**

CDMA based bus system is used to connect peripherals like memory to CPU core. In this implementation a soft core processor, two memory blocks and one parallel port IP is integrated and connected by the designed CDMA bus as Shown in (Fig. 3). A 32 bit system bus is used to inter connect the peripherals with the CPU. The CDMA coded bus reduces the switching activities which imply the reduction of power consumption for on chip communication. This CDMA coded bus system is implemented on data bus and address bus.

**Results and Discussion**

The CDMA bus system is simulated using VHDL in MODELSIM software. the simulation and synthesis report says that CDMA based bus for peripheral communication is reliable and low computation, by decreasing number of lines for on chip peripheral communication it achieves low

![Fig. 1–Principle of the CDMA technique](image1)

![Fig. 2–CDMA Encoding Scheme](image2)

![Fig. 3–System Bus based on CDMA Technique](image3)
power consumption for bus based communication. In this work a 32 bit bus system is implemented for peripheral communication and tested the same using sample data.

Results concerning data transfer over CDMA bus analysed by simulation and synthesis report. From the synthesis report area and latency for various device families are tabulated. (Table 1) shows the result for the encoder and in (Table 2) for the decoder.

Synthesis Report
The CDMA encoding and decoding was implemented in RTL level using VHDL. (Table 1 & 2) shows comparison of device family with logical elements. From the above results, synthesis of CDMA Encoding using Spartan 3E family uses 810 LEs and decoding uses 237 LEs out of 960 LEs available which is 84% and 24% respectively. Vertix5 family uses 276 LEs and 67 LEs out of 1920 available which is 1% and 1% for encoding and decoding respectively. VertixE family uses 869 LEs and 256 LEs out of 6912 available which is 12% and 3% for encoding and decoding respectively. Vertix4 family uses 810 LEs and 233 LEs out of 6144 available which is 13% and 3% for encoding and decoding respectively.

Resource Utilization Report
Resource Utilization meant for utilization of LEs or ALUTs to the FPGA device family. In the performance graph, X-axis represents the device family and Y-axis represents the number of LEs or ALUTs used. The device families are Spartan3E, Vertix4, Vertix5 and Vertix E (XCV600E) is chosen. (Fig. 4a) shows the Resource utilization performance of CDMA Encoding and (Fig. 4b) shows the Resource utilization performance of CDMA Decoding for analyzing these two graphs, Vertix5 family has better resource utilization for both Encoding and Decoding.

Conclusion
The CDMA based on chip communication system is simulated using VHDL in MODEL SIM software. The working of the system is verified by giving test vectors. The FPGA implementation resource utilization, area requirement, latency analysis are done for FPGA family of Spartan3E & Virtex. The best FPGA implementation family is found out thereby. From the above results we can conclude that the best implementation family is Virtex5 due to less area and latency. It is the most advanced, high performance with optimal utilization compare with other family devices.

References


