www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 4 Issue 9 Sep 2015, Page No. 14341-14344

An Overview of Multi-Microcontroller Bus Synchronization for Simultaneous Operation of the Microcontrollers

Anoop Kumar Vishwakarma[1], Neerja Singh[2]

[1]Student (M.Tech.), ECE, ABES Engineering College Ghaziabad, Uttar Pradesh Technical University, Lucknow, India
aanoop88@gmail.com

[2]Assistant Professor, ECE, ABES Engineering College Ghaziabad, Uttar Pradesh Technical University, Lucknow, India
neerja.singh@abes.ac.in

Abstract: ***Now a day's embedded systems are more complex and complicated as compared to the old one. Low power microcontrollers are limited to a specific task. But current time demand is data specific which requires more data processing. Re configurability is also an interesting feature of current time embedded system. So we have to develop a system that is easily re configurable as per our requirement. Cost is also a design factor; we have to develop a system that its cost will not increase. High power microcontrollers are able to process higher data at a higher speed taking higher amount of power. But when our task is limited to power, cost; high power microcontrollers can't be used. High power microcontrollers become problematic for the battery operated embedded systems where power backup is limited. To overcome this problem we are developing Multi-microcontroller System on Programmable Chip (MMSoPC). Synchronization is a key problem in multi-microcontroller systems to operate simultaneously. To synchronize the buses of the microcontrollers, network on chip (NoC) is the solution. Local system buses are connected to a common system bus i.e. global system bus. Global buses are externally available to the outside for different interfacing of various components/peripherals etc. So there is a need of synchronization of local system buses to the global system bus.***

Keywords: MMSoPC, NoC.

1. Introduction

As technology scales toward deep sub-micron, an increasing number of computational units will be integrated onto the same silicon die, posing tight communication requirements on the communication architecture. State-of-the-art on-chip interconnects are shared busses with central arbiters for serializing bus access requests. This simple solution has serious scalability properties, leading to critical performance penalties and energy inefficiencies as the number of integrated cores increases. There is a world-wide ongoing effort to design more scalable communication architectures, for instance leveraging network technology and adapting it to the on-chip scenario. The resulting architectures are known as "Networks-on-Chip" (NoCs) and have some promising features for application to giga-scale systems-on-chip such as high modularity, scalability, potentials or energy savings (e.g., application-specific NoCs) despite the increased design complexity [13].

Current time embedded systems are increasingly based on multiprocessors systems-on-programmable-chip (MPSoPC) [2]. These MPSoPCs typically contain multiple storage elements (SEs), networks (NEs), I/O components, and a number of heterogeneous programmable processors for flexible application support as well as dedicated processing elements (PEs) for achieving high performance and power goals. In order to cope with the design complexity of such systems in a time-efficient way, the abstraction level of the design process has in recent years been raised towards the system level. Design Space Exploration (DSE) is a key ingredient of such system-level design, during which a wide range of design choices are explored, especially during the early design stages. Therefore, such early design choices heavily influence the success or failure of the final product, and can avoid wasting time and effort in further design steps without the possibility of meeting design requirements because of an inappropriate system architecture design.

2. Motivation

Traditionally, scalable performance was requested for high-performance micro-controllers, but now days this request is also common to embedded computing systems. In the recent past, microcontroller designers followed two main trends in order to address this challenge. First, mono-core architectures with advanced techniques to improve their performance were employed. Second, the operating frequency of new generation designs was significantly increased with respect to previous generations. But those design trends have run out of steam [12]. The resulting architectures present power consumption and heat dissipation levels too high to be attractive for commercial products. Also, their complexity is increasing up to the point in which the designs are too complex and error prone, which increases the number and cost of design, re-spins.

As mono-core architectures are ill-suited to solve the above exposed challenges, multi-core architectures have risen as the most elegant solution to address them. Multi-core architectures are composed by several cores that communicate with each other by means of some on-chip interconnection infrastructure, and are usually referred as Systems-on-Chip (SoC). In order to

DOI: 10.18535/ijecs/v4i9.47

exploit the computation scalability provided by multi-core architectures, the communication bottleneck will have to be addressed. Current on-chip interconnects consist mostly of shared arbitrated buses, based on the serialization of bus access requests. The main drawback of this solution is its poor scalability, which will result in unacceptable performance degradation for SoCs of medium or higher complexity (more than a dozen of integrated cores). Moreover, the connection of new blocks to a shared bus increases its associated load capacitance, increasing the energy consumption of bus transactions associated with the broadcast communication paradigm. Thus, Networks-on-Chip (NoCs) emerged as an scalable communication infrastructure that better supports the trend of the SoC paradigm [4, 5, 6]. The basic idea behind NoCs is borrowed from the interconnection networks domain, and envisions the on chip interconnection. NoCs are able to inherit design techniques from interconnection networks, which is a nice quality as interconnection networks are a very mature field. NoCs and interconnection networks impose different constraints. The design process of a SoC consists of a chain of design decisions. Among those decisions, a choice of utmost importance for global system performance is topology selection. A topology describes the connectivity pattern of networked cores, defining hard boundaries to the maximum performance achievable by the NoC as well as influencing other critical parameters like area requirements and power consumption. The most popular NoC topology proposed in the open literature is the 2D-mesh, in which network cores are distributed in a grid-like structure. This topology provides great modularity and a predictable physical layout, but it is known it presents serious scalability issues as the number of network cores connected to it increases. This fact has motivated several works proposing alternative topologies for NoCs [7, 8], as well as techniques to improve the scalability of 2D-meshes by adapting them to the needs of a given design [9], or methodologies to build ad-hoc topologies for application specific designs [10, 9].

3. Generic Architecture of Multi-Microcontroller System

The title Theoretical models of parallel embedded systems are abstracted from the physical models. These models are often used by algorithm designer and VLSI device/Chip developers. The ideal model provides a convenient framework for developing parallel algorithms/hardware without worry about the implementation details or physical constraints [3]. A generic architecture of multi-microcontroller system is shown in fig. 1. There are number of microcontrollers are connected in parallel to a shared bus interface. Input/output sub systems and memory segment are also connected to this shared bus.

4. Internal System Bus Architecture of Multi-Microcontroller System

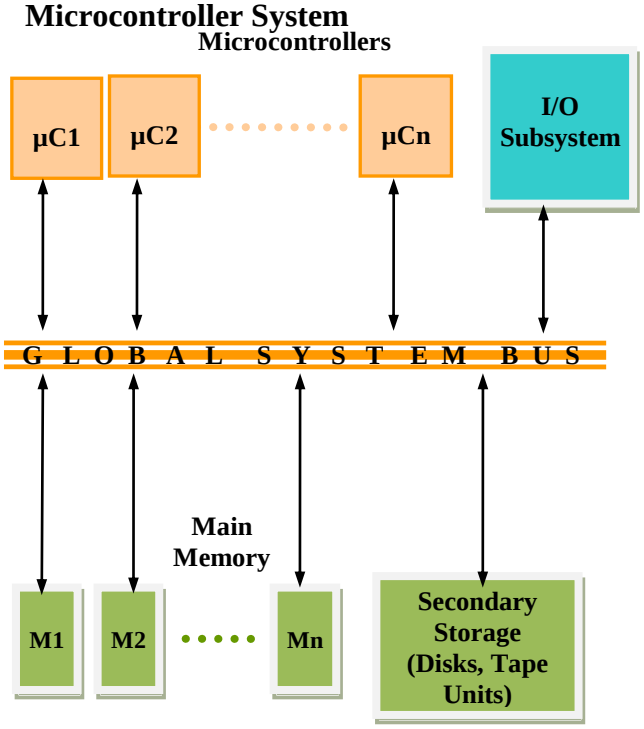

Figure 1: Generic Architecture of Multi-Micro Controller System

If you There is different type of buses inside the multi microcontroller system. They are typically known as LSB (Local System Bus) and GSB (Global System Bus). These two buses basically consist of address buses, data buses, control buses and status buses. A detailed architecture is shown in the fig. 2. GSB is a low cost shared arbitrated bus. In modern buses the most common arbitration solution is a centralized mechanism, implemented as a bus arbiter module. In this mechanism a master device must obtain permission from the arbiter module before starting any transaction. Centralized arbitration has a negative impact over the system performance due to the overhead introduced by the requests to the arbiter module, so the arbitration should be as fast and infrequent as possible.

An example of a commercial shared bus is the AMBA (Advanced Microcontroller Bus Architecture) 2.0 standard, designed for the ARM [1] processor family. This simple bus, which has been widely used in SoC designs, uses the AHB protocol (Advanced High-performance Bus) to interconnect on-chip integrated devices, like ARM processors and RAM memories. It supports differentiated transmission of data and address, both in normal and burst modes. This latter is also utilized to alleviate the overhead introduced by the implemented centralized arbiter. However, the 3.0 version of this same standard supports several simultaneous transactions as well as out-of-order transaction resolution. To summarize, shared buses are the most well-known and the most widely used communications architecture for SoCs, as they are very simple and cost effective. But their scalability is quite limited: it is a viable solution for SoCs with up to five processors and up to ten master devices [11]. For bigger systems it presents unaffordable congestion.

DOI: 10.18535/ijecs/v4i9.47

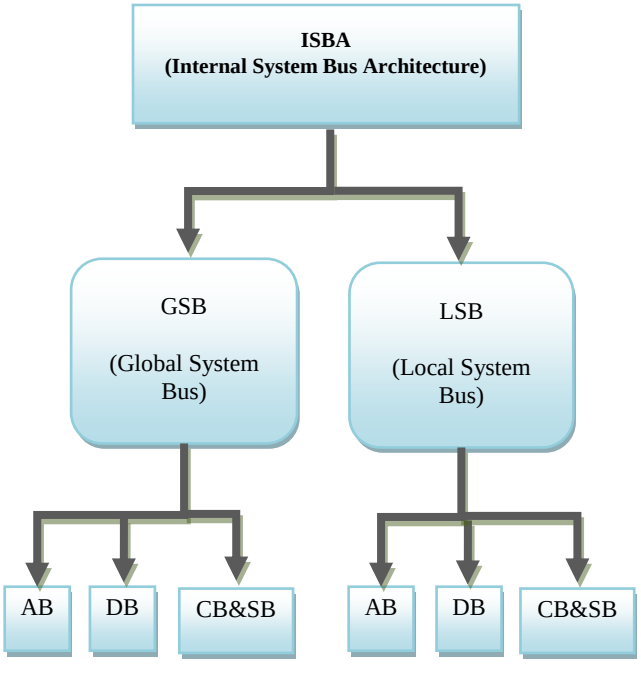

Figure 2: Internal System Bus Architecture of Multi-Microcontroller System

5. Network on Chip Architecture (Proposed Architecture) for the Synchronization of LSB & GSB in Multi-Microcontroller System

To synchronize the local system buses to the global system buses NoC is inserted between the global system bus and local system buses. A clear view of the system with NoC is shown in fig. 3. The operating speed i.e. the switching capability should match with the four microcontrollers. As in our case the number of microcontrollers is four. Networks-on-Chip (NoCs) were proposed as the solution to shared bus scalability issues, emerging as the new parading for designing scalable communication infrastructures in SoCs. The main advantage of a NoC over a shared bus is its capability to reduce design costs while providing high-performance communications. In a NoC, each microcontroller is connected by a Network Interface (NI) to a network link [12]. Those links are point-to-point data lines connected with other links via switches. A link can connect a switch with another switch or with a NI, in such a way that there exists direct or indirect connection between any two cores. Due to the similarities between interconnection networks and NoCs, it is possible to borrow concepts and techniques from the former and apply them to NoCs. Even so, most of the techniques developed for interconnection networks cannot be directly applied in NoCs, while other ones would be completely unaffordable using current technologies. The reason for this lies in the restrictions imposed by each environment. For example, while in interconnection networks buffers are a relatively abundant and affordable resource; in NoCs they are scarce and expensive, with high area requirements and power consumption, which hinders even more their use in NoCs. On the contrary, links are an expensive resource in interconnection networks that must be used with austerity, while in NoCs links are more abundant. Additionally, the total area available and

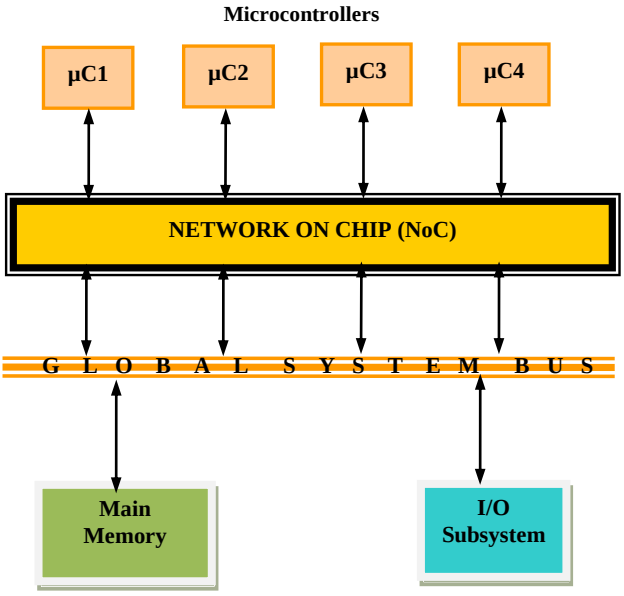

Figure 3: Placing of NoC in Multi-Microcontroller System

the tolerable power consumption budget of a SoC are very limited. An ideal NoC must meet the communications needs of the design while minimizing power and area consumption. This latter fact implies that the switches should be as fast, small and power-efficient as possible, which hinders even more the application of interconnection solutions in NoC designs. The architecture of each NoC component directly or indirectly affects all kinds of design parameters, like maximum operating frequency, buffering requirements, etc. The compilation of architectures of the components that compose the network is referred as network architecture.

6. Proposed Work

In the multi-microcontroller system design the shared bus i.e. global system bus has an important role. As it is the backbone of the multi-microcontroller system. It is clear from the above discussion NoC is the good choice for the synchronization of various local buses (Address, Data, Control and Status) to the global bus (Address, Data, Control and Status). This proposed work for the multi-microcontroller system will be implemented by using HDL. The design and prototyping environment will depend on the system complexity and the design challenge.

7. Conclusions

As per the current demand in the embedded system design there is a need of multi-microcontroller system, in which all microcontroller work simultaneously. The implementation of proposed work i.e. NoC for the multi-microcontroller system on chip for the bus synchronization provides:

(i) An interface between the local buses and global bus.
(ii) Provides capability to operate the microcontrollers simultaneously.
(iii) Buses are accessed under the programmed control.

The most accepted solution to address the above exposed issues consists of simplifying the architecture of microcontrollers. But in order to do this and to keep up with the ever increasing demand of performance, the design should be composed of several simple microcontrollers integrated into a single chip, in what is commonly referred as Systems-on-Chip (SoCs). In this way, a SoC would be composed of several cores with a manageable operating frequency, for instance by implementing simple microcontrollers for generic use and specialized ones for the heaviest tasks, instead of a big complex microcontroller able to carry out the entire task by itself. The resulting architecture will be capable of achieving similar or better raw performance than an equivalent mono-core microcontroller less heat dissipation and power consumption issues.

References

[1] http://www.arm.com

[2] Jia, Z. J., Bautista, T., Nunez, A., Pimentel, A. D., and Thompson, M. 2013. A system-level infrastructure for multidimensional MP-SoC design space co-exploration. ACM Transaction on Embedded Computer System 13, 1s, Article 27 (November 2013).

[3] Kai Hwang (2008). Advance Computer Architecture, Parallelism, Scalability, programmability. McGraw-Hill Companies Inc. New York.

[4] Uday Arun, A. K. Sinha, 2012. A Design Space Exploration of Binary-Tree-Decoder For Multi-Block-External-Memory-Controller in Multi-µc Embedded Soc Design. International Journal of Engineering Inventions, Volume 1, Issue1, August 2012.

[5] L. Benini and G. De Micheli. Networks on chips: a new SoC paradigm. Computer, 35(1):70 –78, 2002.

[6] W.J. Dally and B. Towles. Route packets, not wires: on-chip interconnection networks. In Design Automation Conference, 2001. Proceedings.

[7] Paul Wielage and Kees Goossens. Networks on Silicon: Blessing or Nightmare? In EUROMICRO SYMPOSIUM ON DIGITAL SYSTEM DESIGN (DSD 2002), pages 196–200, 2002.

[8] James Balfour and William J. Dally. Design tradeoffs for tiled CMP onchip networks. In Proceedings of the 20th annual international conference on Supercomputing, ICS '06, pages 187–198, New York, NY, USA, 2006. ACM.

[9] John Kim, James Balfour, and William Dally. Flattened Butterfly Topology for On-Chip Networks. In MICRO '07: Proceedings of the 40th Annual IEEE/ACM International Symposium on Microarchitecture, pages 172–182, Washington, DC, USA, 2007. IEEE Computer Society.

[10] U.Y. Ogras and R. Marculescu. Application-specific network-on-chip architecture customization via long-range link insertion. In Computer-Aided Design, 2005. ICCAD-2005. IEEE/ACM International Conference.

[11] L. Benini. Application specific noc design. In Design, Automation and Test in Europe, 2006. DATE '06. Proceedings.

[12] G. Benini L., De Micheli. Networks on Chips: Technology and Tools. Morgan Kaufmann, 2006.

[13] Francisco Gilabert Villamon, 2011. Design Space Exploration for Networks On-chip. PhD thesis Universitat Polit`ecnica de Val`encia.

[14] Terry Tao Ye, 2003. On-Chip Multiprocessor Communication Network Design and Analysis. PhD thesis, Stanford University.

Author Profile

He did his B. Tech. in Electronics and Telecommunication Engineering from UPTU, Lucknow, M. Tech. (pursuing) in Electronics and Communication Engineering from UPTU, Lucknow. His research interests include Embedded System Design, System-On-Chip Design.

She did her B. Tech. in Electronics and Communication Engineering from Bundelkhand University, Jhansi, and M. Tech. in VLSI Design from UPTU, Lucknow. Her research interests include VLSI Design.