International Journal of Science and Research (IJSR)
ISSN: 2319-7064
ResearchGate Impact Factor (2018): 0.28 | SJIF (2018): 7.426

Survey of Efficient NOC Router Designs and Programming Model for Embedded Application

Trupti Patil[1], Dr. Anuradha S[2]

[1, 2]Lingaraj Appa Engineering College, Bidar, Guru Nanak Dev Engineering College, Bidar

Abstract: ***System on chips containing IP cores and traditional methods for communication such as bus, are not suitable solution for future System on chips (SOC). As the complexity of the SOC is increasing, it is impossible to send signals from one end to another end within a clock cycle. Problems such as global wire delay and global synchronization will be the limitations; Network on chip is an emerging approach for the implementation of on chip communication architecture. Network on chip a communication centric approach and it is a possible solution for communication architecture of future System on chips that are composed of switches and IP cores where communicate among each other through switches.. In contrast to normal beliefs, on chip interconnections suffer from certain physical limitations which lead to great performance reduction. How the changes made in traditional NOC to best fit for the today's requirement is the subject of this paper. We discuss new techniques. We discuss the unique problems posed by synchronous NOCs and discuss the different Asynchronous NOC model as the promising solution. We survey work to build accurate simulation models for on chip communication, propose a programming model for efficient router design for embedded application.***

Keywords: System on chip (SOC), IP cores, Synchronous, Asynchronous, Network on chip (NOC)

1. Introduction

Traditionally, ICs have been designed with dedicated point-to-point connections, with one wire dedicated to each signal. Now with new developments in chip manufacturing technologies several Intellectual Property (I.P.) blocks such as processor cores, memories, dedicated hardware can be built on single chip with high increase in computation performance. For such rising computation performance the communication bandwidth requirement also increases with same rate.

To solve the problem of the traditional bus in area, scaling and power consumption, etc., a new on-chip communication structure Network-on-Chip has been proposed. NoC provides high performance communication at the cost of ar increase in the structure complexity. Networks on chips (NoCs). Large, complex multiprocessor-based SoC platforms are already well into existence, and, according tc common expectations and technology roadmaps, the emergence of billion-transistor chips is just around the corner. The complexity of such systems calls for a serious revisiting of several on-chip communication issues. In this special issue, we focus on an emerging paradigm that effectively addresses and presumably can overcome the many on-chip interconnection and communicatior challenges that already exist in today's chips or will likely occur in future chips

In section II we discuss literature survey of the network or chip designs. In section III design for an efficient router is proposed finally conclusion is drawn in section IV

2. Literature Review

Complex VLSI IC design has been revolutionized by the widespread adoption of the SoC paradigm. SoC designs consist of one or more IPs, designed for a single or narrow set of applications with a highly charecterizable communication. As the level of a chip integration continues to advances at a fast pace, the desire for efficient interconnects rapidly increases. Busses have successfully been implemented in virtually all complex System on Chip (SoC) Silicon designs, have typically been handcrafted around either a specific set of features relevant to a narrow target market, or support for a specific processor. Several trends have forced evolutions of systems architectures, in turn driving evolutions of required busses. These trends are:
Application convergence: The mixing of various traffic types in the same SoC design (Video, Communication, Computing and etc.). These traffic types, although very different in nature, for example from the Quality of Service point of view, must now share resources that were assumed to be "private" and handcrafted to the particular traffic in previous designs.

Moore's law is driving the integration of many IP Blocks in a single chip. This is an enabler to application convergence, but also allows entirely new approaches (parallel processing on a chip using many small processors) or simply allows SoCs to process more data streams (such as communication channels)

Consequences of silicon process evolutions between generations: Gates cost relatively less than wires, both from an area and performance perspective, than a few years ago.
Time-To-Market pressures are driving most designs to make heavy use of synthesizable RTL rather than manual layout, in turn restricting the choice of available implementation solutions to fit a bus architecture into a design flow. These trends have driven of the evolution of many new bus architectures. These include the introduction of split and retry techniques, removal of tri-state buffers and multi-phase-clocks, introduction of pipelining, and various attempts to define standard communication sockets.

The most popular bus architectures utilize hierarchical levels of buses. For example, Core Connect has three levels of hierarchy: Processor Local Bus (PLB), On-chip Peripheral Bus (OPB), and Device Control Register (DCR). PLB

provides a high performance and low latency processor bus with separate read and write transactions, while OPB provides low speed with separate read and write data buses to reduce bottlenecks caused by slow I/O devices such as serial ports, parallel ports, and UARTs. The daisy chained DCR offers a relatively low-speed data path for passing status and configuration information.

The user configurable Triscend bus architecture utilizes a bus FIFO to enhance bus pipelining between masters and slaves [1]. The arbiter logic is relatively simple because the FIFO is both the single master for the slave side and also the single slave for the master side. The FIFO, however, requires additional memory and makes it difficult to predictably satisfy real-time constraints as compared to prioritized buffers. The Silicon Backplane from Sonic Inc. guarantees fixed bandwidth and latency by Time Division Multiplexed Access (TDMA) based arbitration [2]

Issues: Bus latency
Busses must be over-designed in bandwidth to reduce their utilization rate Special mechanisms such as pre-scheduled or time multiplexed transactions must be devised to reduce conflicts. While these mechanisms are sometimes used for memory access scheduling in support of real-time flows, they are rarely found in traditional busses. Strict TDMA techniques trade arbitration latency for minimized transport latency, since bursts will only use a fraction of the bus aggregate bandwidth. As a consequence, complex schemes must be devised to optimize both behaviors. crossbars or multilayered busses are used in place of shared busses. This limits conflicts to transactions directed to the same target, traditional crossbars still mix transaction, transport and physical layers in a way similar to traditional busses, they present only partial solutions. They continue to suffer the following: Scalability: IP block reusability Maximum frequency, wire congestion and area: Crossbars do not isolate transaction handling from transport Crossbar control logic is complex, datapaths are heavily loaded and very wide

Area: Traditional busses have been perceived as very area efficient because of their shared nature. As we already discussed, this shared nature drives both operation frequency and system performance scalability down. Some techniques have been introduced in recent busses to fix these issues: Pipelining added to sustain bus frequencies: with busses having typically more than 100 wires, each pipeline stage costs at least 1Kgates FIFOs inserted to deal with arbitration latency: Even worse, to sustain throughput as latency grows, buffers must be inserted in the bridges between the inter-cluster and cluster-level busses.[3]

Using on-chip interconnection networks in place of ad-hoc global wiring structures the top level wires on a chip and facilitates modular design. With this approach, system modules (processors, memories, peripherals, etc...) communicate by sending packets to one another over the network. The structured network wiring gives well-controlled electrical parameters that eliminate timing iterations and enable the use of high-performance circuits to reduce latency and increase bandwidth. The area overhead required to implement an on-chip network is modes design-specific global on-chip wiring with a general-purpose on-chip interconnection networa chip employing an on-chip network is composed of a number of network clients: processors, DSPs, memories, peripheral controllers, gateways to networks on other chips, and custom logic. Instead of connecting these top-level modules by routing dedicated wires, they are connected to a network that routes packets between them.

Using an on-chip interconnection network to replace top-level wiring has advantages of structure, performance, and modularity. A network structures the top-level wires simplifying their layout and giving them well-controlled electrical parameters. These well controlled electrical parameters in turn enable the use of high-performance circuits that result in significantly lower power dissipation, higher propagation velocity, and higher bandwidth that is possible with conventional circuits [4]

As we discussed advantage of NOC over bus in terms of power, modularity, scalability but the problem now arise is that, as the System-on-chip (SoC) designs integrate a variety of cores and I/O interfaces, which usually operate at different clock frequencies. Communication between un clocked and clock domains requires careful synchronization, which inevitably introduces metastability and some uncertainty in timing. Thus, any chip with multiple clock domains is already globally asynchronous.

The literature is rife with techniques to handle the integration of multiple clock domains, most of which rely on a localized clock domain- crossing circuit that lets one clock domain talk directly to another. Most designs still implement long-range communication with synchronous circuits, requiring a widely distributed clock, which can be challenging to implement at high frequency so one solution is a globally asynchronous, locally synchronous (GALS). The concept of quasi-delay-insensitive (QDI) timing model, 3 which requires that the circuit function correctly regardless of any gate delay and most wire delays.

The asynchronous system scales linearly with utilization all the way down to zero power at 0 percent utilization. The synchronous system also scales linearly with utilization (assuming it holds the old data value for a padding cycle), but the clock load on the latches consumes a constant amount of power.

The synchronous system also has a data dependent power dissipation that varies widely, because the power also scales linearly with activity, unlike in asynchronous systems. Asynchronous on-chip networks are power efficient and tolerant to process variation but they are slower than synchronous on-chip networks. Network-on-chip [4] is the state-of-the-art on-chip communication fabric for current multi-processor SoC systems.

The on-chip network could be a synchronous network where routers are driven by a global clock, or an asynchronous network where routers are self-timed circuits connected by asynchronous pipelines. Thanks to mature EDA tools and the timing assumptions allowed by the global clock, synchronous networks are fast and area efficient but the clock tree is power consuming [2]. By contrast, the clock-

less asynchronous networks are comparatively slow but power efficient. In addition, they are tolerant to process variation and could divide the whole chip into several isolated clock domains, which unifies the network interface and shortens the overall design time. Although asynchronous networks tend to be slow, their advantages are crucial to Nanoscale SoC systems.

The problem associated with asynchronous can be solved with the design which utilizes two novel techniques: channel slicing and the look ahead pipeline; therefore, all other design aspects are set to broadly accepted configurations *Channel slicing*: The state-of-the-art quasi delay-insensitive (QDI) pipelines in routers are built by synchronizing multiple bit-level pipelines (sub-channels) the wormhole flow control low latency asynchronous router has been implemented [5].

To take the advantage of both synchronous and asynchronous design designs based on Globally asynchronous and locally synchronous designs are proposed to further enhance the performance of on-chip communications of Globally Asynchronous Locally Synchronous Systems (GALS), a dynamic reconfigurable multi-synchronous router architecture is proposed to increase network on chip (NoC) efficiency by changing the path of the communication link in the runtime traffic situation. In order to address GALS issues and bandwidth requirements, multi-synchronous bidirectional NoC's router is developed and it guarantees higher packet consumption rate, better bandwidth utilization with lower packet delivery latency. The consensus is that current techniques, when extrapolated to future technologies, will face significant shortcomings in several key areas. In addition to the latency throughput are predicted to become significant bottlenecks for system performance.

The first contribution is two new highly-concurrent asynchronous network building blocks, or "primitives," to support the routing and arbitration functions of the network. Each component is carefully designed for high performance and low area and power overheads, using a *transition-signaling*, i.e., two-phase, communication protocol , which has only one roundtrip communication per channel per transaction. In principle, transition-signaling is a preferred match for high performance asynchronous systems yet it presents major practical design challenges: most existing two-phase asynchronous pipeline components are complex, with large latency, area and power overheads. to further reduce area overheads. Additional network level features, such as the quality of service, are also under consideration, following the advances in. Finally, alternative variants of the asynchronous NoC will be considered using delay-insensitive encodings, such as level-encoded transition signaling codes, which provide greater timing-robustness than bundled data at the expense of coding efficiency. As system-level interconnect incurs increasing penalties in latency, round-trip cycle time and power, and as timing-variability becomes an increasing design challenge, there is renewed interest in using two-phase delay-insensitive asynchronous protocols for robust system-level communication.

However, in practice, it is extremely inefficient to build local asynchronous computation nodes with two-phase logic, hence four-phase (i.e., return-to-zero) computation blocks are typically used, two new architecture for a family of asynchronous protocol converters that translate between two- and four-phase protocols, thus facilitating robust system design using efficient global two-phase communication and local four-phase computation. These converters facilitate the design of systems-on-chip using robust global two-phase delay-insensitive communication and efficient local asynchronous four-phase protocols for function blocks.

With small modifications, the above converters can easily be designed between LEDR and other widely-used four-phase asynchronous logic styles, such as 1-of-4 and single-rail bundled data, as shown in [9]. Several recent approaches to heterogeneous system design [10], [11], [12] use similar asynchronous architectures with local function blocks and global communication networks between blocks. Each could potentially benefit from the proposed LEDR conversion scheme [13]

The asynchronous routers and links have been designed based on a four-phase protocol in a quasi-delay-insensitive (QDI) logic style for robustness to voltage and process variations. In contrast, a two-phase protocol [5] is used for a high-throughput asynchronous communication link, because the asynchronous operations are done in a single roundtrip of handshaking, instead of two in the four-phase protocol. However, complex latches and function blocks are required in the two-phase protocol, which leads to area- and delay-inefficient computation blocks. Therefore, the computation blocks, such as the router, are designed based on the four-phase protocol. the high-throughput protocol converter based on an independent encoding/decoding scheme is proposed for robust asynchronous communication in the GALS-NoC architecture, where the asynchronous routers and links are designed based on four- and two-phase protocols in the QDI logic style, respectively. Since the two-phase input and output signals are independently encoded to and decoded from the four-phase signals, respectively, each conversion is completely performed without the input-output dependency.

A new delay-insensitive data encoding scheme for global communication, called *level encoded transition signaling (LETS)*, is introduced. LETS is a generalization of LEDR encoding. In LEDR, only one of two wires changes value per data bit per transaction. In contrast, in LETS, only one of N = 2n (1-of-N) wires changes value per n data bits per transaction. Hence, LEDR can be regarded as a special case: 1-of-2 LETS codes. Compared to existing non-return-to-zero schemes (LEDR), higher-dimension LETS codes have a potential power advantage, with significantly reduced switching activity per data bit. Compared to most common return-to-zero encoding schemes], LETS also has potential power and throughput advantages, since fewer rails switch per transaction and no return-to-zero phase is required.[14]

The LEDR protocol has two potential benefits over returnto-zero (RZ) schemes for asynchronous global communication [15]: *throughput* and *power*. Unlike return-to-zero schemes,

no `spacer' or reset phase is required, hence LEDR provides a significant system-level throughput advantage. Furthermore, LEDR can provide a power advantage, since only one transition occurs on a rail per data bit transmission, while return-to-zero schemes require two transitions. These benefits have encouraged recent applications using LEDR encoding [16, 17].

3. Proposed Design

In this paper, a high-throughput and compact asynchronous NoC router based on the LEDR encoding with a novel packet structure constraint is proposed for highly reliable NoCs. In the proposed NoC, the LEDR encoding is used for both communication links and routers. A processing core partitions a packet into an even number of flits that are transferred through communication links and routers. Since each flit is represented based on the two-phase encoding, which consists of two kinds of phase information (ODD and EVEN), the phase information of header and tail flits is uniquely determined. Thus, the router can be implemented without considering the phase information, significantly reducing the complexity of the LEDR encoding. As a result,

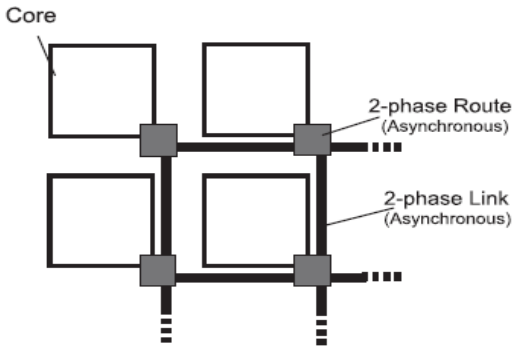

Figure 1: An Asynchronous NoC based on LEDR encoding

The proposed NoC is to benefit at a maximum from the LEDR encoding that the communication steps and the number of signals representing a packet becomes half in comparison with the four-phase encoding. The reduction of the number of signals would lead to a small chance of collisions between flits of different packets compared with the four-phase encoding under the same traffic patterns, leading to high-throughput data communication.

Fig. 2 shows the overall structure of the proposed asynchronous NoC router, which consists of five input units and five output units. The input unit includes two-stage Pipelatches (PLs), shifter (SH), and routing controller (RC). The output unit includes two-stage PLs, arbitration controller (AC), and multiplexer (MX). Each input unit is connected to other four output units except its corresponding output unit. This router has five input and output ports. Flits are transmitted from one port to one of other ports. Every signal is a two phase signal except sel signals for PLs. Initially, the first and the last-stage PLs are transparent.

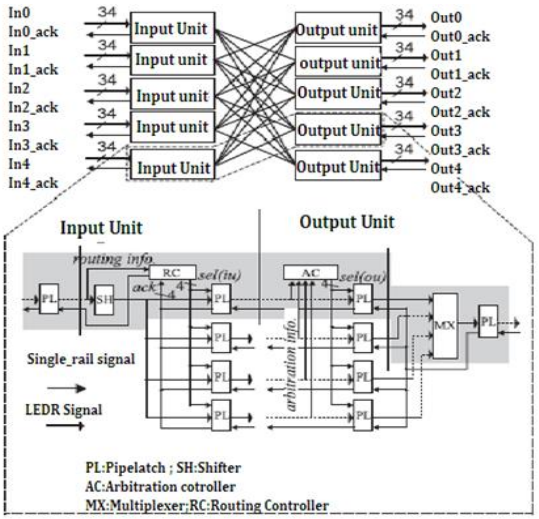

Figure 2: The proposed Asynchronous router

The router operates based on a three-stage pipeline manner. The operation depends on the flit type. A header flit determines the routing path in the router. First, a header flit is processed in the RC, and the destination port of the packet is determined. The phase type of the header flit is always ODD in our router. In the SH, the address information of a header flit is shifted to eliminate the first subaddress information, which was already used in the RC. Then, the AC determines which flit can be transferred to an output unit. When multiple flits simultaneously request to use the same output unit, a flit selected by the AC is transferred, while the other flits remain. The output unit selects a flit in MX and then transfers the flit to the other routers via communication links or the processing core connected to the router. Body flits are simply transferred through the routing path determined by a header flit in the router. A tail flit is processed in the RC and resets the destination of the packet. Then, the tail flit releases the AC. After the tail flit is transferred to the other router or the processing core, one of the other flits can use the same output unit.

4. Conclusion

In this paper we have discussed advantage of network on chip over traditional shared architecture, here synchronous design over asynchronous noc designs are compared and also discussed improvement of NOC design to best fit the present requirements along with their advantage and limitations. After studying the different NOC design a high-throughput compact delay insensitive asynchronous NoC router based on LEDR encoding with a packet-structure constraint is proposed. Since a routing computation in this design is performed by using only single-phase information, the hardware complexity of two-phase encoding is alleviate with maintaining timing robustness. Thus, the proposed NoC is to benefit at a maximum from the two phase encoding that communication steps and the number of signals being used become half in comparison with the four-phase encoding

References

[1] S. Winegarden, “Bus Architecture of a System on a Chip with User Configurable System Logic,” IEEE

Journal of Solid State Circuits, March 2000, Vol. 35, No. 3, pp. 425-433.
[2] D. Wingard and A. Kurosawa, "Integration Architecture for System-on-a-Chip Design," Proceedings of IEEE 1998 Custom Integrated Circuits Conference, May 1998, pp. 85-88.
[3] A comparison of Network-on-Chip and Busses by Arteris(www.arteris.com)
[4] William J. Dally and Brian Towles "Route Packets, Not Wires: On-Chip Interconnection Networks" Computer Systems Laboratory Stanford University Stanford, CA 94305 {billd,btowles}@cva.stanford.edu
[5] Lines, A. -- Asynchronous interconnect for synchronous SoC design IEEE Micro Volume 24 issue 1 2004 [doi 10.1109%2Fmm.2004.1268991]
[6] A. Hemani, T. Meincke, S. Kumar, A. Postula, T. Olsson, P. Nilsson, J. Oberg, P. Ellervee, and D. Lundqvist, "Lowering power consumption in clock by using globally asynchronous locally synchronous design style," in *Proc. of DAC*, 1999, pp. 873–878.
[7] Wi Song and Doug Edwards "A Low Latency Wormhole Router for Asynchronous On-chip Networks"School of Computer Science, University of Manchester Manchester, M13 9PL UK"
[8] Michael N. Horak, Steven M. Nowick, -"A Low-Overhead Asynchronous Interconnection Network for GALS Chip Multiprocessors '', *IEEE*, Matthew Carlberg, *Student Member, IEEE*, and Uzi Vishkin, *Senior Member, IEEE*
[9] A. Mitra, W. F. McLaughlin, and S. M. Nowick, "Efficient asynchronous protocol converters for two-phase delay-insensitive global communication," in *Proc. 13th IEEE Int. Symp. Asynchronous Circuits Syst.*, 2007, pp. 185–186.
[10] E. Beigne and P. Vivet, "Design of on-chip and off-chip interfaces for a GALS NoC architecture," in *Proc. 12th IEEE Int. Symp. Asynchronous Circuits Syst.*, 2006, pp. 172–181.
[11] R. Dobkin, R. Ginosar, and A. Kolodny, "Fast asynchronous shift register for bit serial communication," in *Proc. 12th IEEE Int. Symp. Asynchronous Circuits Syst.*, 2007, pp. 117–127.
[12] N. Karaki, "Asynchronous design: An enabler for flexible microelectronics (keynote talk)," in *Proc. 12th IEEE Int. Symp. Asynchronous Circuits Syst.*, 2006, p. xii [Online]. Available: http://tima.imag.fr/conferences/ ASYNC
[13] William F. McLaughlin, Amitava Mitra, and Steven M. Nowick –"Asynchronous Protocol Converters for Two-Phase Delay-Insensitive Global Communication" IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 17, NO. 7, JULY 2009
[14] Peggy B. McGee, Melinda Y. Agyekum, Moustafa A. Mohamed and Steven M. Nowick –"A Level-Encoded Transition Signaling Protocol for High-Throughput Asynchronous Global Communication" Department of Computer Science
[15] A. Mitra, W. F. McLaughlin, and S. M. Nowick. Effcient asynchronous protocol converters for two-phase delay-insensitive global communication. In *Proc. of the 13th IEEE International Symposium on Asynchronous Circuits and Systems*, pages 186.185, 2007. Columbia UniversityvNew York, NY 10027
[16] R. R. Dobkin, R. Ginosar, and A. Kolodny. High rate wave-pipelinedvasynchronous on-chip bit-serial data link. In *Proc. of the 13th IEEEvInternational Symposium on Asynchronous Circuits and Systems*, pages3.14, 2007.
[17] B. R. Quinton, M. R. Greenstreet, and S. J. E. Wilton. Asynchronous ic interconnect network design and implementation using a standard asic_ow. In *Proc. of the 2005 International Conference on Computer Design (ICCD 2005)*, pages 267.274, 2005.