Argo: A Real-Time Network-on-Chip Architecture With an Efficient GALS Implementation
In this paper, we present an area-efficient, globally asynchronous, locally synchronous network-on-chip (NoC) architecture for a hard real-time multiprocessor platform. The NoC implements message-passing communication between processor cores. It uses statically scheduled time-division multiplexing (TDM) to control the communication over a structure of routers, links, and network interfaces (NIs) to offer real-time guarantees. The area-efficient design is a result of two contributions: 1) asynchronous routers combined with TDM scheduling and 2) a novel NI microarchitecture. Together they result in a design in which data are transferred in a pipelined fashion, from the local memory of the sending core to the local memory of the receiving core, without any dynamic arbitration, buffering, and clock synchronization. The routers use two-phase bundled-data handshake latches based on the Mousetrap latch controller and are extended with a clock gating mechanism to reduce the energy consumption. The NIs integrate the direct memory access functionality and the TDM schedule, and use dual-ported local memories to avoid buffering, flow-control, and synchronization. To verify the design, we have implemented a 4 x 4 bitorus NoC in 65-nm CMOS technology and we present results on area, speed, and energy consumption for the router, NI, NoC, and postlayout.

General information
State: Published
Organisations: Department of Applied Mathematics and Computer Science, Embedded Systems Engineering, Eindhoven University of Technology
Authors: Kasapaki, E. (Intern), Schoeberl, M. (Intern), Sørensen, R. B. (Intern), Müller, C. (Intern), Goossens, K. (Ekstern), Sparse, J. (Intern)
Pages: 479-492
Publication date: 2016
Main Research Area: Technical/natural sciences

Publication information
Journal: IEEE Transactions on Very Large Scale Integration (VLSI) Systems
Volume: 24
Issue number: 2
ISSN (Print): 1063-8210
Ratings:
BFI (2018): BFI-level 2
Web of Science (2018): Indexed yes
BFI (2017): BFI-level 2
An Asynchronous Time-Division-Multiplexed Network-on-Chip for Real-Time Systems

Multi-processor architectures using networks-on-chip (NOCs) for communication are becoming the standard approach in the development of embedded systems and general purpose platforms. Typically, multi-processor platforms follow a globally asynchronous locally synchronous (GALS) timing organization. This thesis focuses on the design of Argo, a NOC targeted at hard real-time multi-processor platforms with a GALS timing organization.

To support real-time communication, NOCs establish end-to-end connections and provide latency and throughput guarantees for these connections. Argo uses time division multiplexing (TDM) in combination with a static schedule to implement virtual end-to-end circuits. TDM is a straightforward way to provide guarantees and to share the resources efficiently, and it has an efficient hardware implementation. Argo supports a GALS system organization, and additionally it explores more flexible timing within its structure, to address signal distribution issues, using a network of synchronous routers.

NOCs consist of a switching structure of routers connected by links, with network interfaces (NIs) that connect the processors to the switching structure. Argo uses a novel NI design that supports time-predictability, and asynchronous
routers that form a time-elastic network. The NI design integrates the DMA functionality and the TDM schedule, and uses dual-ported local memories. The routers combine the router functionality and asynchronous elastic behavior. They also use a gating mechanism to reduce the energy consumption. The combination of the NI design and the router design supports the formation of end-to-end paths in the NOC, from the local memory of a sending core to the local memory of a receiving core. These end-to-end paths do not require any dynamic arbitration, buffering, flow control, or clock synchronization, in the routers or the NIs.

This thesis explores the implementation of the individual components of Argo, as well as several complete instances of the Argo NOC. The implementations target both FPGA technology and 65 nm CMOS technology. It is shown that (i) the NI design is scalable and four to five times smaller than previously published NIs for similar NOCs, (ii) the router design is power efficient and two to three times smaller than equivalent router designs, and (iii) the overall Argo NOC is around four times smaller than other TDM NOCs. Argo is an important part of the T-CREST platform and used in a number of configurations.

The flexible timing organization of Argo combines asynchronous routers with mesochronous NIs, which are connected to individually clocked cores, supporting a GALS system organization. The mesochronous NIs operate at the same frequency, possibly with some skew, while the network of asynchronous routers absorbs this skew within certain limits. The elasticity of the asynchronous network is explored, answering the question of how much skew the Argo NOC can absorb. A qualitative analysis studies the parameters affecting the elasticity and its limits. A quantitative analysis models the Argo behavior using timed-graph models and worst-case timing separation of events analysis to evaluate the elasticity of Argo. The results show that the skew absorbed by the network of routers can be two or more cycles, depending on the frequency applied at its endpoints, the NIs.

Overall this thesis presents the design and implementation of Argo, and the analysis of its elastic behavior. It shows that Argo provides hard real-time guarantees in a straightforward way, it has an efficient implementation and it is time-elastic.

General information
State: Published
Organisations: Department of Applied Mathematics and Computer Science, Embedded Systems Engineering
Authors: Kasapaki, E. (Intern), Sparsø, J. (Intern), Schoeberl, M. (Intern)
Number of pages: 150
Publication date: 2015

Publication information
Place of publication: Kgs. Lyngby
Publisher: Technical University of Denmark (DTU)
Original language: English

Series: DTU Compute PHD-2015
Number: 361
ISSN: 0909-3192
Main Research Area: Technical/natural sciences
Electronic versions:
phd361_Kasapaki_E.pdf
Publication: Research › Ph.D. thesis – Annual report year: 2015

T-CREST: Time-predictable multi-core architecture for embedded systems
Real-time systems need time-predictable platforms to allow static analysis of the worst-case execution time (WCET). Standard multi-core processors are optimized for the average case and are hardly analyzable. Within the T-CREST project we propose novel solutions for time-predictable multi-core architectures that are optimized for the WCET instead of the average-case execution time. The resulting time-predictable resources (processors, interconnect, memory arbiter, and memory controller) and tools (compiler, WCET analysis) are designed to ease WCET analysis and to optimize WCET performance. Compared to other processors the WCET performance is outstanding. The T-CREST platform is evaluated with two industrial use cases. An application from the avionic domain demonstrates that tasks executing on different cores do not interfere with respect to their WCET. A signal processing application from the railway domain shows that the WCET can be reduced for computation-intensive tasks when distributing the tasks on several cores and using the network-on-chip for communication. With three cores the WCET is improved by a factor of 1.8 and with 15 cores by a factor of 5.7. The T-CREST project is the result of a collaborative research and development project executed by eight partners from academia and industry. The European Commission funded T-CREST.

General information
State: Published
The Argo NOC: Combining TDM and GALS

Argo is a network-on-chip developed for use in a multi-core platform designed specifically for hard real-time applications and it supports message passing across virtual end-to-end channels. Argo implements these channels using time-division-multiplexing (TDM) of the resources in the NOC following a static schedule. This requires some form of global synchrony across the platform. At the same time it is generally accepted that a large chip should employ some form of globally-asynchronous locally-synchronous (GALS) organization. By using asynchronous routers and by rethinking the microarchitecture of the network interfaces we have managed to combine TDM and GALS and obtain a very hardware-efficient implementation of the NOC. The paper gives a brief overview of the Argo NOC and focuses on two important issues: how to safely bring the NOC out of reset and timing analysis of the network of asynchronous routers.

A Loosely Synchronizing Asynchronous Router for TDM-Scheduled NOCs

This paper presents an asynchronous router design for use in time-division-multiplexed (TDM) networks-on-chip. Unlike existing synchronous, mesochronous and asynchronous router designs with similar functionality, the router is able to silently skip over cycles/TDM-slots where no traffic is scheduled and hence avoid all switching activity in the idle links and router ports. In this way switching activity is reduced to the minimum possible amount.

The fact that this relaxed synchronization is sufficient to implement TDM scheduling represents a contribution at the conceptual level. The idea can only be implemented using asynchronous circuit techniques. To this end, the paper explores the use of “click-element” templates. Click-element templates use only flipflops and conventional gates, and this greatly simplifies the design process when using conventional EDA tools and standard cell libraries. Few papers, if any, have explored this.
Argo: A Time-Elastic Time-Division-Multiplexed NOC using Asynchronous Routers

In this paper we explore the use of asynchronous routers in a time-division-multiplexed (TDM) network-on-chip (NOC), Argo, that is being developed for a multi-processor platform for hard real-time systems. TDM inherently requires a common time reference, and existing TDM-based NOC designs are either synchronous or mesochronous. We use asynchronous routers to achieve a simpler, smaller, and more robust, self-timed design. Our design exploits the fact that pipelined asynchronous circuits also behave as ripple FIFOs. Thus, it avoids the need for explicit synchronization FIFOs between the routers. Argo has interesting elastic timing properties that allow it to tolerate skew between the network interfaces (NIs).

The paper presents Argo NOC-architecture and provides a quantitative analysis of its ability of absorb skew between the NIs. Using a signal transition graph model and realistic component delays derived from a 65nm CMOS implementation, a worstcase analysis shows that a typical design can tolerate a skew of 1-5 cycles (depending on FIFO depths and NI clock frequency). Simulation results of a 2 x 2 NOC confirm this.

Synthesis and Layout of an Asynchronous Network-on-Chip using Standard EDA Tools

Asynchronous circuit design is well understood but design tools supporting asynchronous design are largely lacking, and designers are limited to using conventional EDA-tools. These tools have a built-in synchronous mind-set and this complicates their use for asynchronous implementation. One example is the key role that clock signals play in specifying time-constraints for the synthesis. In this paper explain how we handled the synthesis and layout of an asynchronous network-on-chip for a multi-core platform. Focus is on the design process while the actual NOC-design and its performance are presented elsewhere.
An area-efficient network interface for a TDM-based Network-on-Chip

Network interfaces (NIs) are used in multi-core systems where they connect processors, memories, and other IP-cores to a packet switched Network-on-Chip (NOC). The functionality of a NI is to bridge between the read/write transaction interfaces used by the cores and the packet-streaming interface used by the routers and links in the NOC. The paper addresses the design of a NI for a NOC that uses time division multiplexing (TDM). By keeping the essence of TDM in mind, we have developed a new area-efficient NI micro-architecture. The new design completely eliminates the need for FIFO buffers and credit based flow control - resources which are reported to account for 50--85% of the area in existing NI designs. The paper discusses the design considerations, presents the new NI micro-architecture, and reports area figures for a range of implementations.

Router Designs for an Asynchronous Time-Division-Multiplexed Network-on-Chip

In this paper we explore the design of an asynchronous router for a time-division-multiplexed (TDM) network-on-chip (NOC) that is being developed for a multi-processor platform for hard real-time systems. TDM inherently requires a common time reference, and existing TDM-based NOC designs are either synchronous or mesochronous, but both approaches have their limitations: a globally synchronous NOC is no longer feasible in today's sub micron technologies and a mesochronous NOC requires special FIFO-based synchronizers in all input ports of all routers in order to accommodate for clock phase differences. This adds hardware complexity and increases area and power consumption. We propose to use asynchronous routers in order to achieve a simpler, more robust and globally-asynchronous NOC, and this represents an unexplored point in the design space. The paper presents a range of alternative router designs. All routers have been synthesized for a 65nm CMOS technology, and the paper reports post-layout figures for area, speed and energy and compares the asynchronous designs with an existing mesochronous clocked router. The results show that an asynchronous router is 2 times smaller, marginally slower and with roughly the same energy consumption, while offering a robust solution to the clock distribution problem. The paper further explores "clock-gating" of the individual pipeline stages in the asynchronous routers, and shows that this can lead to significant power savings.
A Statically Scheduled Time-Division-Multiplexed Network-on-Chip for Real-Time Systems

This paper explores the design of a circuit-switched network-on-chip (NoC) based on time-division-multiplexing (TDM) for use in hard real-time systems. Previous work has primarily considered application-specific systems. The work presented here targets general-purpose hardware platforms. We consider a system with IP-cores, where the TDM-NoC must provide directed virtual circuits - all with the same bandwidth - between all nodes. This may not be a frequent scenario, but a general platform should provide this capability, and it is an interesting point in the design space to study. The paper presents an FPGA-friendly hardware design, which is simple, fast, and consumes minimal resources. Furthermore, an algorithm to find minimum-period schedules for all-to-all virtual circuits on top of typical physical NoC topologies like 2D-mesh, torus, bidirectional torus, tree, and fat-tree is presented. The static schedule makes the NoC time-predictable and enables worst-case execution time analysis of communicating real-time tasks.

General information
State: Published
Organisations: Department of Informatics and Mathematical Modeling, Embedded Systems Engineering, Computer Science and Engineering
Authors: Schoeberl, M. (Intern), Brandner, F. (Intern), Sparsø, J. (Intern), Kasapaki, E. (Intern)
Pages: 152-160
Publication date: 2012

Projects:
Asynchronous Network-on-Chip for Time-Predictable Multi-Core Embedded Systems
Department of Applied Mathematics and Computer Science
Period: 01/10/2011 → 19/06/2015
Number of participants: 6
Phd Student: Kasapaki, Evangelia (Intern)
Supervisor: Schoeberl, Martin (Intern)
Main Supervisor: Sparsø, Jens (Intern)
Examiner: Nannarelli, Alberto (Intern)
Jantsch, Axel (Ekstern)
Yakovlev, Alexandre (Ekstern)

**Financing sources**
Source: Internal funding (public)
Name of research programme: Institut, samfinansiering
Project: PhD