Chapter 13
A High-Performance and Low-Power On-Chip Network with Reconfigurable Topology

Mehdi Modarressi
Sharif University of Technology, Iran

Hamid Sarbazi-Azad
Sharif University of Technology, Iran

ABSTRACT
In this chapter, we present a reconfigurable architecture for network-on-chips (NoC) on which arbitrary application-specific topologies can be implemented. The proposed NoC can dynamically tailor its topology to the traffic pattern of different applications, aiming to address one of the main drawbacks of existing application-specific NoC optimization methods, i.e. optimizing NoCs based on the traffic pattern of a single application. Supporting multiple applications is a critical feature of an NoC as several different applications are integrated into the modern and complex multi-core system-on-chips and chip multiprocessors and an NoC that is designed to run exactly one application does not necessarily meet the design constraints of other applications. The proposed NoC supports multiple applications by configuring as a topology which matches the traffic pattern of the currently running application in the best way. In this chapter, we first introduce the proposed reconfigurable topology and then address the two problems of core to network mapping and topology exploration. Experimental results show that this architecture effectively improves the performance of NoCs and reduces power consumption.

INTRODUCTION
With the advance of the semiconductor technology in recent years, current multi-core system-on-chips (SoCs) have grown in size and complexity and future SoCs will consist of complex integrated components communicating with each other at very high-speed rates. Moreover, the microprocessor industry is moving from single-core to multi-core and eventually to many-core architectures, containing tens to hundreds of identical cores arranged as chip multiprocessors (CMPs) [Owens et al. 2007]. The lack of scalability in bus-based systems and large area overhead and unpredictability of electrical parameters of point-to-point dedicated links have motivated researchers to propose packet-
switched Network-on-Chip (NoC) architectures to overcome complex on-chip communication problems [Benini and De Micheli 2002; Guerrier and Greiner 2000; Dally and Towles 2001; Jantsch and Tenhunen 2003].

However, it has been shown that, in future technologies (especially 22 nm), the power consumption of the current NoCs, when providing the required performance of typical CMP and multi-core SoC applications, is about 10 times greater than the power budget that can be devoted to them [Owens et al. 2007]. Application-specific optimization is one of the most effective approaches to bridge this exiting gap between the current and the ideal NoC power/performance metrics. This class of optimization methods tries to customize the architecture of an NoC for a target application, where the application and its traffic characteristics are known at design time. Most state-of-the-art NoC architectures and their design flows provide design-time NoC optimization (including topology selection and mapping) for a single application [Bjerregaard and Mahadevan 2006]. In other words, they try to generate an optimized NoC based on the traffic pattern of a single application and then, synthesize the chip. However, today’s (often programmable) multi-core SoCs are highly complex and cost-effective chips, and as technology advances, it becomes more cost-effective to integrate several different applications onto a single SoC chip. As a result, NoC architectures should closely match the traffic characteristics and performance requirements of different applications. Since different applications have different functionalities, the inter-core communication characteristics can be very different across the applications. Consequently, an NoC that is designed to run exactly one application does not necessarily meet the design constraints of other applications. Prior work [Kim et al. 2008] shows, by conducting simulation over 1500 different NoC configurations (topology, buffer size, and bit-width), that no single NoC provides optimal performance across a range of applications.

In this chapter, we tackle this problem for two effective application-specific optimization methods, i.e. topology generation and core to network mapping, by introducing a reconfigurable NoC architecture designed based on a regular mesh-based NoC. It enables the network topology to dynamically match the communication pattern of the currently running application. Afterwards, with a set of different applications as input, we develop a two-phase algorithm which first maps all of the IP-cores used by the applications onto the reconfigurable NoC nodes and then implements a suitable topology for each input application by appropriately configuring the NoC.

Optimizing the network topology and core to network mapping are two application-specific NoC customization methods which dramatically affect the network performance-related characteristics such as average inter-core distance, total wire length, and communication flow distributions. These characteristics, in turn, determine power consumption and average network latency of NoC architectures. Topology determine the connectivity of the NoC nodes while mapping determines on which node each processing core should be physically placed. Mapping algorithms generally try to place the processing cores communicating more frequently near each other; when the number of routers between two cores reduces, the power consumption and delay of their communications decreases linearly.

The reconfiguration of the proposed architecture is achieved by inserting several simple switches in the network which allow the network to dynamically change the inter-node connections and implement the topology that best matches the running application demands. More precisely, we try to reduce the network hops (or number of routers) between the source and destination nodes of high volume communication traces (or ideally connect them directly) by bypassing one or more intermediate routers. This can lead to considerable performance improvement since the power/latency of the router pipeline stages has a