Abstract

Network on Chips[1][2][3][4] are critical elements of modern System on Chip(SoC) as well as Chip Multiprocessor (CMP) designs. Network on Chips (NoCs) help manage high complexity of designing large chips by decoupling computation from communication. SoCs and CMPs have a multiplicity of communicating entities like programmable processing elements, hardware acceleration engines, memory blocks as well as off-chip interfaces. With power having become a serious design constraint[5], there is a great need for designing NoC which meets the target communication requirements, while minimizing power using all the tricks available at the architecture, microarchitecture and circuit levels of the design. This thesis presents a holistic, QoS based, power optimal design solution of a NoC inside a CMP taking into account link microarchitecture and processor tile configurations.

Guaranteeing QoS by NoCs involves guaranteeing bandwidth and throughput for connections and deterministic latencies in communication paths. Label Switching based Network-on-Chip (LS-NoC) uses a centralized LS-NoC Management framework that engineers traffic into QoS guaranteed routes. LS-NoC uses label switching, enables bandwidth reservation, allows physical link sharing and leverages advantages of both packet and circuit switching techniques. A flow identification algorithm takes into account bandwidth available in individual links to establish QoS guaranteed routes. LS-NoC caters to the requirements of streaming applications where communication channels are fixed over the lifetime of the application. The proposed NoC framework inherently supports heterogeneous and ad-hoc SoC designs.

A multicast, broadcast capable label switched router for the LS-NoC has been designed, verified, synthesized, placed and routed and timing analyzed. A 5 port, 256

bit data bus, 4 bit label router occupies 0.431 mm^2 in 130nm and delivers peak bandwidth of 80Gbits/s per link at 312.5MHz. LS Router is estimated to consume 43.08 mW. Bandwidth and latency guarantees of LS-NoC have been demonstrated on streaming applications like HiperLAN/2 and Object Recognition Processor, Constant Bit Rate traffic patterns and video decoder traffic representing Variable Bit Rate traffic. LS-NoC was found to have a competitive $\frac{\text{Area}\times\text{Power}}{\text{Throughput}}$ figure of merit with state-of-the-art NoCs providing QoS. We envision the use of LS-NoC in general purpose CMPs where applications demand deterministic latencies and hard bandwidth requirements.

Design variables for interconnect exploration include wire width, wire spacing, repeater size and spacing, degree of pipelining, supply, threshold voltage, activity and coupling factors. An optimal link configuration in terms of number of pipeline stages for a given length of link and desired operating frequency is arrived at. Optimal configurations of all links in the NoC are identified and a power-performance optimal NoC is presented. We presents a latency, power and performance trade-off study of NoCs using link microarchitecture exploration. The design and implementation of a framework for such a design space exploration study is also presented. We present the trade-off study on NoCs by varying microarchitectural (e.g. pipelining) and circuit level (e.g. frequency and voltage) parameters.

A System-C based NoC exploration framework is used to explore impacts of various architectural and microarchitectural level parameters of NoC elements on power and performance of the NoC. The framework enables the designer to choose from a variety of architectural options like topology, routing policy, etc., as well as allows experimentation with various microarchitectural options for the individual links like length, wire width, pitch, pipelining, supply voltage and frequency. The framework also supports a flexible traffic generation and communication model. Latency, power and throughput results using this framework to study a 4x4 CMP are presented. The framework is used to study NoC designs of a CMP using different classes of parallel computing benchmarks[6].

One of the key findings is that the average latency of a link can be reduced by increasing pipeline depth to a certain extent, as it enables link operation at higher link frequencies.

There exists an optimum degree of pipelining which minimizes the energy-delay product of the link. In a 2D Torus when the longest link is pipelined by 4 stages at which point least latency (1.56 times minimum) is achieved and power (40% of max) and throughput (64% of max) are nominal. Using frequency scaling experiments, power variations of up to 40%, 26.6% and 24% can be seen in 2D Torus, Reduced 2D Torus and Tree based NoC between various pipeline configurations to achieve same frequency at constant voltages. Also in some cases, we find that switching to a higher pipelining configuration can actually help reduce power as the links can be designed with smaller repeaters. We also find that the overall performance of the ICNs is determined by the lengths of the links needed to support the communication patterns. Thus the mesh seems to perform the best amongst the three topologies (Mesh, Torus and Folded Torus) considered in case studies.

The effects of communication overheads on performance, power and energy of a multiprocessor chip using L1, L2 cache sizes as primary exploration parameters using accurate interconnect, processor, on-chip and off-chip memory modelling are presented. On-chip and off-chip communication times have significant impact on execution time and the energy efficiency of CMPs. Large caches imply larger tile area that result in longer inter-tile communication link lengths and latencies, thus adversely impacting communication time. Smaller caches potentially have higher number of misses and frequent of off-tile communication. Energy efficient tile design is a configuration exploration and trade-off study using different cache sizes and tile areas to identify a power-performance optimal configuration for the CMP.

Trade-offs are explored using a detailed, cycle accurate, multicore simulation framework which includes superscalar processor cores, cache coherent memory hierarchies, on-chip point-to-point communication networks and detailed interconnect model including pipelining and latency. Sapphire, a detailed multiprocessor execution environment integrating SESC, Ruby and DRAMSim was used to run applications from the Splash2 benchmark (64K point FFT). Link latencies are estimated for a 16 core CMP simulation on Sapphire. Each tile has a single processor, L1 and L2 caches and a router. Different sizes of L1 and L2 lead to different tile clock speeds, tile miss rates and tile area and hence

interconnect latency.

Simulations across various L1, L2 sizes indicate that the tile configuration that maximizes energy efficiency is related to minimizing communication time. Experiments also indicate different optimal tile configurations for performance, energy and energy efficiency. Clustered interconnection network, communication aware cache bank mapping and thread mapping to physical cores are also explored as potential energy saving solutions. Results indicate that ignoring link latencies can lead to large errors in estimates of program completion times, of up to 17%. Performance optimal configurations are achieved at lower L1 caches and at moderate L2 cache sizes due to higher operating frequencies and smaller link lengths and comparatively lesser communication. Using minimal L1 cache size to operate at the highest frequency may not always be the performance-power optimal choice. Larger L1 sizes, despite a drop in frequency, offer a energy advantage due to lesser communication due to misses.

Clustered tile placement experiments for FFT show considerable performance per watt improvement (1.2%). Remapping most accessed L2 banks by a process in the same core or neighbouring cores after communication traffic analysis offers power and performance advantages. Remapped processes and banks in clustered tile placement show a performance per watt improvement of 5.25% and energy reduction of 2.53%. This suggests that processors could execute a program in multiple modes, for example, minimum energy, maximum performance.