SoC Design
Lecture 13: NoC
(Network-on-Chip)

Department of Computer Engineering
Sharif University of Technology
Outline

- SoC Interconnect
- NoC – Introduction
- NoC layers
- Typical NoC Router
- NoC Issues
  - Switching
  - Performance evaluation
    - Power consumption
  - Different topologies of NoC
  - Routing Algorithms
- Summary
Building a CMP with Shared Memory

- Build a Chip Multi-Processor (CMP) with existing modules
- Local & Shared Memory architecture
- Schema example:
Approaches of Interconnect

- Dedicated wiring
  - poor reusability
  - poor scalability
  - problems of wiring latency and noise

- Shared bus
  - limited bandwidth
  - limited system complexity
Bus Inheritance

From Board level into Chip level…
Typical Solution: Bus

Segmented Bus

Shared Bus

Diagram showing segmented bus with a bus symbol B.
Typical Solution: Bus

Original bus features:
- One transaction at a time
- Central Arbiter
- Limited bandwidth
- Synchronous
- Low cost

New features:
- Versatile bus architectures
- Pipelining capability
- Burst transfer
- Split transactions
- Transaction preemption and resume
- Transaction reordering…
Approaches of Interconnect (Cont’d)

- Parallel topologies have been proposed to increase the amount of delivered bandwidth
  - Ex. partial or full crossbars
  - Scalability limitations of crossbar-based interconnection fabrics are well known

- New communication protocols have been developed: more effective exploitation of the available bandwidth
  - Ex. AMBA 3.0 AXI and the open-core protocol (OCP)
  - Provide support for point-to-point communication only and do not provide any specification on the interconnect fabric
Approaches of Interconnect (Cont’d)

- Networks-on-chip (NoCs)
  - Most important alternative for the design of modular and scalable communication architectures
  - Providing inherent support to the integration of heterogeneous cores through standard socket interfaces
  - Relieve system-level integration issues
  - Suitable to deal with the challenges of nanoscale technology

- Area and power overheads is significant in spite of the performance benefits
Outline

- SoC Interconnect
- NoC – Introduction
- NoC layers
- Typical NoC Router
- NoC Issues
  - Switching
  - Performance evaluation
    - Power consumption
  - Different topologies of NoC
  - Routing Algorithms
- Summary
New Solution: On-chip Communication

- **Bus based interconnect**
  - Low cost
  - Easier to Implement
  - Flexible

- **Networks on Chip**
  - Layered Approach
  - Buses replaced with Networked architectures
    - Better electrical properties
    - Higher bandwidth
    - Energy efficiency
    - Scalable
What is NoC?

- According to Wikipedia:
  - “Network-on-a-chip (NoC) is a new paradigm for System-on-Chip (SoC) design. NoC based-systems accommodate multiple asynchronous clocking that many of today's complex SoC designs use. The NoC solution brings a networking method to on-chip communications and claims roughly a threefold performance increase over conventional bus systems.”

- Imprecise...
NoC Exemplified
Basic Ingredients of an NoC

- N Computational Resources
  - Processing Elements (PE)
- 1 Connection Topology
- 1 Routing technique
- $M \leq N$ Switches
- N Network Interfaces
For the Connoisseurs…

- 1 Addressing system
- 1 Switch-level Arbitration policy
- 1 Communication Protocol
- 1 Programming model
  - Message passing
  - Shared Memory
NoC’s Requirements

Requirements:

- Different QoS must be supported
  - Bandwidth
  - Latency
- Distributed deadlock free routing
- Distributed congestion/flow control
- Low VLSI Cost
NoC: Good news

😊 Only point-to-point one-way wires are used, for all network sizes
😊 Aggregated bandwidth scales with the network size
😊 Routing decisions are distributed and the same router is re-instantiated, for all network sizes
😊 NoCs increase the wires utilization (as opposed to ad-hoc p2p wires)
NoC: Bad news

- Internal network contention causes (often unpredictable) latency
- The network has a significant silicon area
- Bus-oriented IPs need smart wrappers
- Software needs clean synchronization in multiprocessor systems
- System designers need reeducation for new concepts
Facts about NoCs

- It is a way to decouple computation from communication
- The design is *layered* (physical, network, application…)
- Communication between *processing elements* in NoC takes place by encapsulating data in *packets*
- The elementary packet piece to which switch and routing operations apply is the *flit*
Network on Chip vs. Bus

Networks are preferred over buses:

- Higher bandwidth
- Concurrency, effective spatial reuse of resources
- Higher levels of abstraction
- Modularity - Design Productivity Improvement
- Scalability
NoC vs. “Off-Chip” Networks

What is Different?

- Routers on Planar Grid Topology
- Short p2p Links between routers
- Unique VLSI Cost Sensitivity:
  - Area-Routers and Links
  - Power
NoC vs. “Off-Chip” Networks (Cont’d)

- No legacy protocols to be compliant with …
- No software → simple and hardware efficient protocols
- Different operating env. (no dynamic changes and failures)
- Custom Network Design – You design what you need!

Example1: Replace modules
Who first had the idea?

- No clear parenthood. The most referred papers according to Google (#cit.)
  - Guerrier’00 (204), *A Generic Architecture for On-Chip Packet-Switched Interconnections*
  - Dally’01 (392), *Route Packets, Not Wires: On-Chip Interconnection Networks*
  - Benini’02 (417), *Networks on Chips: A New SoC Paradigm*
  - Kumar’02 (184), *A Network on Chip Architecture and Design Methodology*
SPIN (Guerrier et al., DATE ’00/’03)

- Wormhole switching, adaptive routing and credit-based flow control
- It is based on a fat-tree topology
- A flit is only one word (36 bits, 4 bits are for packet framing)
- The input buffers have a depth of 4 words
Dally et al., DAC’01

- 2D folded torus topology
- Wormhole routing and Virtual Channels (VC)
Kumar et al., ISLVLSI’02

- Chip-Level Integration of Communicating Heterogeneous Elements, CLICHÉ’
- 2D Mesh Topology
- Message Passing

Sharif University of Technology

SoC: Network On Chip
Butterfly Fat Tree
- Wormhole, Virtual channels
- Header flits: 3 ck cycles latency (input arbitration, routing, output arbitration)
- “Body” flits: 3 ck cycles (input arbitration, switch traversal, output arbitration)
Both VCT and WH, GT and BE
GT uses TDM to avoid contention and create virtual circuits
   - In each time slot a block of 3 flits is transferred from In “j” to Out “k” in a S&F fashion
BE uses Matrix Scheduling
GT connections set up by BE special system packets
Prototype with WH
   - 5 ports
   - 0.13 um, 0.26 mm², 500/166 MHz
   - Flit size = 3 words, each 32 bits
   - 80 Gb/s aggregate bandwidth
Common Properties

- Data integrity: means that data is delivered uncorrupted
- Lossless data delivery: means no data is dropped in the interconnect
- In-order data delivery: specifies that the order in which data is delivered is the same order in which it has been sent
- Throughput and Latency: services that offer time related bounds
The Interconnection Network

Network on Chip (NOC)

CPU → Network Interface → Switch → Switch → Network Interface → Mem.

Buffer regions within the Network Interface and Switches.

Links to/from other switches or network interfaces.
Communication Centric Design

Application

Architecture Library

Architecture / Application Model

NoC Optimization

Configure

Evaluate

Analysis / Profile

Good?

No

Synthesis

Refine

Optimized NoC
Outline

- SoC Interconnect
- NoC – Introduction
- NoC layers
  - Physical layer
  - Data link layer
  - Network layer
  - Transport Layer
  - Application Layer
- Typical NoC Router
- NoC Issues
  - Switching
  - Performance evaluation
    - Power consumption
  - Different topologies of NoC
  - Routing Algorithms
  - Region
- Summary
Network on Chip - Layers

- Software
- Transport
- Network
- Wiring

Traffic Modeling
Architectures
Queuing Theory

Separation of concerns

Networking
Several Layers of Communication

- Communication Layers and unit of communication:
  - Physical layer: Word
  - Data link layer: Flit
  - Network layer: Packet
  - Transport layer: Message
  - Application layer
Flow of Data from Source to Sink through the NoC Components
NoC Layering – Physical interpretation
Physical Layer

- Parameters:
  - Physical distance
  - Number of lines
  - Activity control
  - Buffers and pipelining

![Diagram of physical layer with switches and wires](image)
Data Link Layer

- Parameters:
  - Line frequency versus switch frequency (word versus flit)
  - Buffering
  - Error correction
  - Power optimization; e.g. avoid activity and power optimized encoding
Network Layer

- Parameters:
  - flit size versus packet size
  - Network address scheme, e.g. 4 + 4 bit for 16*16 resources
  - Routing algorithm
  - Priority classes: e.g. 2 classes:
    - high priority, fixed delay flits
    - low priority, best effort delay flits
  - Error correction
Transport Layer

- Parameters:
  - Message size
  - Virtual channels with traffic profiles
  - Signaling
  - Priority classes of channels, e.g.
    - constant bit rate traffic
    - varying bit rate traffic
  - Network resource management
  - Error correction
Application Layer

- Interprocess communication at the task level:
  - send / receive for individual messages
  - open; write/read; close for channel based communication

- Mapping issues:
  - Assigning tasks to resources
  - Translating task addresses to resource/task addresses
  - Establishing and closing channels
  - Static allocation versus dynamic allocation
Outline

- SoC Interconnect
- NoC – Introduction
- NoC layers
- Typical NoC Router

- NoC Issues
  - Switching
  - Performance evaluation
    - Power consumption
  - Different topologies of NoC
  - Routing Algorithms
  - Region

- Summary
Regular Network on Chip
Typical NoC Router
Outline

- SoC Interconnect
- NoC – Introduction
- NoC layers
- Typical NoC Router
- NoC Issues
  - Switching
  - Performance evaluation
    - Power consumption
  - Different topologies of NoC
  - Routing Algorithms
- Summary
NoC Issues

- Application Specific Optimization
  - Switching
  - Buffers
  - Routing
  - Topology
  - Mapping to topology
  - Implementation and Reuse
Mapping Problems

- Link bandwidth requirement
- Communication latency
  - routing algorithm
- Communication flow
  - power consumption
Switching

- Again, techniques inherited from Computer and Communication Networks
- Flow control is a synchronization protocol for transmitting and receiving a unit of information
- Switching techniques differ in the relationship between the sizes of the physical and message flow control units

- New constraints in silicon: area and power
  - Use as few buffers as possible
- Store & Forward and Virtual-Cut-Through
  - Need buffers size for an entire packet, unsuited!
- Limited buffer size in
  - Wormhole
- Virtual channels
  - Increase buffer size…
Switching (Cont’d)

- Circuit Switching
- Packet Switching (Store-and-Forward)
- Virtual Cut-Through Switching
- Wormhole switching
- Hybrid architecture
Circuit Switching

- A physical path from the source to the destination is reserved prior to the transmission of the data
  - routing information set up during initialization
- Guaranteed transmission latency and throughput
- Switch design has lower complexity
- Advantageous when messages are infrequent and long
  - Suitable for application-specific SoC
- Disadvantage: physical path is reserved for the duration of the message and may block other messages
Circuit Switching (Cont’d)
Packet Switching

- Each packet is individually routed from source to destination
- A packet is completely buffered at each intermediate node before it is forwarded to the next node
- Advantageous when messages are short and frequent
- Disadvantage:
  - Splitting a message into packets produces some overhead
  - In addition to the time required at source and destination nodes, every packet must be routed at each intermediate node
- Structure
  - Switch design has higher complexity
- Modularity
  - interface
  - reusability
- Scalability
  - bandwidth
Packet Switching (Cont’d)
Virtual Cut-Through Switching

- Packet switching: a packet must be received in its entirety before any routing decision can be made and the packet forwarded to the destination.
- Virtual Cut-Through: Packet header can be examined as soon as it is received.
  - Router starts forwarding the header and following data bytes as soon as routing decisions have been made and the output buffer is free.
NoC Wormhole Routing

- Message packets are also pipelined through the network
- The *flit* is the unit of message flow control, and input and output buffers at a router are typically large enough to store a few flits
- Wormhole Routing
  - For reduced buffering

Wormhole Packet:

<table>
<thead>
<tr>
<th>Flit (routing info)</th>
<th>Flit</th>
<th>Flit</th>
<th>Flit</th>
<th>Flit</th>
</tr>
</thead>
</table>

![Wormhole Packet Diagram]
NoC Wormhole Router
Virtual Channel

- Sharing of a physical channel by several logically separate channels with individual and independent buffer queues

- Advantages:
  - Avoiding deadlocks
  - Optimizing wire utilization
  - Improving performance
  - Providing differentiated services

- Disadvantages:
  - Area overhead
  - Power overhead
Virtual Channel (Cont’d)
Outline

- NoC Issues
  - Switching
  - Performance evaluation
    - Power consumption
  - Different topologies of NoC
  - Routing Algorithms
NoC Analysis

- Universally applicable parameters of NoC:
  - Latency
  - Bandwidth
  - Jitter
  - Power consumption
  - Area usage
NoC Analysis (Cont’d)

- Analytical performance model for the on-chip communications
- Stochastic modeling based on queuing theory
Importance of Performance Evaluation

- Customization of NoC resources
- Reduction of cost in the communication network
- Maintaining the required Quality of Service
- Evaluation of different configuration to increase performance
Latency Metrics

Total Latency = Sender Overhead + Time of Flight + Message Size ÷ BW + Receiver Overhead
Sources of Power Consumption

- Shared memory power consumption
- Node power consumption
- Interconnect network power consumption
  - The internal node switch
  - The internal buffers
  - The interconnect wires

\[ P_{NoC} = P_{routers} + P_{links} \]
Simple Energy Model

- Hu assume:
  - $E_{\text{bit}} = E_{S\text{bit}} + E_{B\text{bit}} + E_{W\text{bit}} + E_{L\text{bit}}$

- Simplifying assumptions:
  - Buffer implemented using latches and flip-flops
  - Negligible internal wire energy

- Router to Router Energy (minimal routing)
  - $E_{\text{bit}} = n_{\text{hops}} \times E_{S\text{bit}} + (n_{\text{hops}} - 1) \times E_{L\text{bit}}$
  - $n_{\text{hops}}$ proportional to energy consumption
Outline

- NoC Issues
  - Switching
  - Performance evaluation
  - Different topologies of NoC
  - Routing Algorithms
Different Topologies for NoC

- Heritage of networks with new constraints
  - Need to accommodate interconnects in a 2D layout
  - Cannot route long wires (clock frequency bound)

a) SPIN,
b) CLICHE’
c) Torus
d) Folded torus
e) Octagon
f) BFT.
Topologies - Examples

Fat tree

Bus

Sharif University of Technology
SoC: Network On Chip
Topologies – Examples (Cont’d)

- **Tree**
  - 7 CPUs with local memory
  - 1 Global memory
  - 1 8-port switch
  - 8 Network interfaces

- **Star**
  - 7 CPUs with local memory
  - 1 Global memory
  - 1 8-port switch
  - 8 Network interfaces
Topology - Mesh

- Bidirectional links (double the connections)
- Asymmetric at edges
Topology – Mesh (Cont’d)

- Resource-to-switch ratio: 1
- A switch is connected to 4 switches and 1 resource
- A resource is connected to 1 switch
- Max number of hops grows with $2n$
Topologies - Tree

- One route
- Bidirectional links
- Top-level nodes overloaded
Outline

- NoC Issues
  - Switching
  - Performance evaluation
  - Different topologies of NoC
  - Routing Algorithms
Routing

- Objective: Find a path from a source node to a destination node on a given topology
- One of the key components that determine the performance of the network
- Performance measures of a routing algorithm:
  - Reduce the number of hops and overall latency
  - Balance the load of network channels
Routing Algorithm

- Connection-oriented vs. connectionless
  - Connection-oriented: involve a dedicated (logical) connection path established prior to data transport.
  - Connectionless: the communication occurs in a dynamic manner with no prior arrangement between the sender and the receiver

- Minimal vs. nonminimal routing
  - Minimal: only consider minimal routes (shortest path)
  - Non-Minimal: allow even nonminimal routes
Routing Algorithm (Cont’d)

- Central vs. distributed control
  - Centralized control: routing decisions are made globally
  - Distributed control: routing decisions are made locally

- Delay vs. loss
  - Delay model: datagrams are never dropped
  - Loss model: datagrams can be dropped

- Deterministic vs. adaptive routing
Deterministic Routing

- *Deterministic* algorithms always choose the same path between two nodes

- Advantages
  - Simple and inexpensive to implement
  - Usual deterministic routing is minimal, which leads to short path length
  - Packets arrive in order

- Disadvantage
  - Lack of path diversity can create large load imbalances
Dimension-Order Routing in Tori and Meshes

- Also called e-cube routing
- Digits of destination address are used to route the packet through the network
- Since a torus can be traversed in clockwise or counterclockwise direction, the preferred direction in each dimension have to be calculated first
- The packet is the routed along these directions \{ +x, -x, +y, -y \} until it reaches its destination

Advantages
- Simple Router (no tables, simple logic)
- Power efficient communication
- No deadlock scenario
Oblivious Routing

- *Oblivious* algorithms always choose a route without knowing about the state of the networks state
  - All random algorithms are oblivious algorithms
  - All deterministic algorithms are oblivious algorithms
Adaptive Routing

- *Adaptive* algorithms use information about the state of the network to make routing decisions
  - Length of queues
  - Historical channel load

- Typically a node has only local information such as information on the load of the connected links

- Two types:
  - Fully adaptive
  - Partially adaptive
Fully Adaptive Routing

- Fully-Adaptive Routing does not restrict packets to take the shortest path
- This can help to avoid congested areas
- Fully-Adaptive Routing may result in deadlock
- Mechanisms must be added to prevent deadlock
Simulation Issues

- **Stochastic traffic generators**
  - Ease of implementation/simulation
  - Fast simulation
  - Self-similar traffic used by some

- **Trace-Based Simulation**
  - Need for extensive pre-simulation
  - Long simulations (days-weeks)
  - Accurate results
Traffic Pattern

Categories of traffic within a system:

1. Latency Critical: with stringent latency demands such as for critical interrupts, memory access, etc.
   - These often have low payload

2. Data Streams: Have high payload and demand QoS in terms of bandwidth
   - Most often it is large, mostly fixed bandwidth, which may be jitter critical
     - Ex. : MPEG data, DMA access, etc

3. Miscellaneous: with no specific requirements of commitment from the network
Applications

- Main NoC feature: high communication bandwidth
- Desirable feature for MP-SoC: low communication latency
- The twos are often contrasting requirements:
  - "Bandwidth problems can be cured with money. Latency problems are harder because the speed of light is fixed—you can’t bribe God." — Anonymous
- Desperately seeking benchmarks and killer applications
  - Multimedia?
What is New?

- Amazing application of network ideas to the chip context
- But ideas need to be re-contextualized
- Old constraints
  - Latency, bandwidth
- New constraints are very tight
  - Area, Power, Clocks
- Differences of fine-grain NoC with large-grain Networks
  - Today links are 100% reliable. Might become false for ultra-scaled technologies and globally asynchronous NoC
  - For many applications, lowest latency is more important than highest bandwidth
Summary

- In SoC: Interconnect dominates delay and power
- Design challenges for future SoC design
  - Wiring delay
  - Synchronization
  - Noise
  - Power dissipation
- On chip communication with a network is very simple and reliable
- NoC architecture has a great effect on delay and power of SoC interconnection
- Routing algorithm, switching mechanism, topology and traffic pattern affect both power and delay of an NoC