Lecture 25:
Networks & Interconnect
Networks on a Chip

Prepared by: Professor David A. Patterson
Professor Bill Dally (Stanford)
Edited and presented by: Prof. Jan Rabaey
Computer Science 252, Spring 2000

Project Presentations Next Week

• The Good News: No presentations on Tu
  faculty retreat – time for contemplation
• The Bad News: A monster session on Th: 2-6pm (?)
• Each group gets 15 minutes (12’ + 3) – Use
  overhead or mail in powerpoint a day in
  advance
• Conference style presentation
• Be concise
**Review: Interconnections**

- Communication between computing elements
- Protocols to cover normal and abnormal events
- Performance issues: HW & SW overhead, interconnect latency, bisection BW
- Media sets cost, distance
- HW and SW Interface to computer affects overhead, latency, bandwidth

**Review: Performance Metrics**

<table>
<thead>
<tr>
<th>Sender</th>
<th>Transmission time (size ÷ bandwidth)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(processor busy)</td>
</tr>
<tr>
<td>Time of Flight</td>
<td>Transport Latency</td>
</tr>
<tr>
<td>Receiver</td>
<td>Receiver Overhead (processor busy)</td>
</tr>
</tbody>
</table>

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

Includes header/trailer in BW calculation?
Interconnect Issues

- Performance Measures
- Interface Issues
- Network Media
- Connecting Multiple Computers

Connecting Multiple Computers

- Shared Media vs. Switched: pairs communicate at same time: "point-to-point" connections
- Aggregate BW in switched network is many times shared
  - point-to-point faster since no arbitration, simpler interface
- Arbitration in Shared network?
  - Central arbiter for LAN?
  - Listen to check if being used ("Carrier Sensing")
  - Listen to check if collision ("Collision Detection")
  - Random resend to avoid repeated collisions; not fair arbitration;
  - OK if low utilization

(Aka data switching interchanges, multistage interconnection networks, interface message processors)
Example Interconnects

<table>
<thead>
<tr>
<th>Interconnect</th>
<th>MPP</th>
<th>LAN</th>
<th>WAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example</td>
<td>CM-5</td>
<td>Ethernet</td>
<td>ATM</td>
</tr>
<tr>
<td>Maximum length</td>
<td>25 m</td>
<td>500 m;</td>
<td>copper: 100 m optical: 2 km—25 km</td>
</tr>
<tr>
<td>Number data lines</td>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Clock Rate</td>
<td>40 MHz</td>
<td>10 MHz</td>
<td>&lt; 155.5 MHz</td>
</tr>
<tr>
<td>Shared vs. Switch</td>
<td>Switch</td>
<td>Shared</td>
<td>Switch</td>
</tr>
<tr>
<td>Maximum number of nodes</td>
<td>2048</td>
<td>254</td>
<td>&gt; 10,000</td>
</tr>
<tr>
<td>Media Material</td>
<td>Copper</td>
<td>Twisted pair copper wire or Coaxial cable</td>
<td>Twisted pair copper wire or optical fiber</td>
</tr>
</tbody>
</table>

Switch Topology

- Structure of the interconnect
- Determines
  - Degree: number of links from a node
  - Diameter: max number of links crossed between nodes (max dist)
  - Average distance: number of hops to random destination
  - Bisection: minimum number of links that separate the network into two halves (worst case)
- Warning: these three-dimensional drawings must be mapped onto chips and boards which are essentially two-dimensional media
  - Elegant when sketched on the blackboard may look awkward when constructed from chips, cables, boards, and boxes (largely 2D)
### Important Topologies

<table>
<thead>
<tr>
<th>N</th>
<th>Type</th>
<th>Degree</th>
<th>Diameter</th>
<th>Ave Dist</th>
<th>Bisection</th>
<th>Bisection Diam</th>
<th>Ave D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1024</td>
<td>1D mesh</td>
<td>&lt; 2</td>
<td>N/3</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2D mesh</td>
<td>&lt; 4</td>
<td>2(N^{1/2} - 1) / 3</td>
<td>N^{1/2}</td>
<td>63</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3D mesh</td>
<td>&lt; 6</td>
<td>3(N^{1/3} - 1) / 3</td>
<td>N^{2/3}</td>
<td>-30</td>
<td>-10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>nD mesh</td>
<td>&lt; 2n</td>
<td>n(N^{1/n} - 1) / 3</td>
<td>N^{(n-1)/n}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ring</td>
<td>2</td>
<td>N/2</td>
<td>N/4</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2D torus</td>
<td>4</td>
<td>N^{1/2}</td>
<td>N^{1/2} / 2</td>
<td>32</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td></td>
<td>k-ary n-cube</td>
<td>2n</td>
<td>n(N^{1/n})</td>
<td>nN^{n/2}</td>
<td>15</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(N = k^n)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hypercube</td>
<td>n</td>
<td>n = LogN</td>
<td>n/2</td>
<td>10</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cube-Connected Cycles</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Topologies (cont)

<table>
<thead>
<tr>
<th>N</th>
<th>Type</th>
<th>Degree</th>
<th>Diameter</th>
<th>Ave Dist</th>
<th>Bisection</th>
<th>Bisection Diam</th>
<th>Ave D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1024</td>
<td>2D Tree</td>
<td>3</td>
<td>2Log_2 N</td>
<td>~2Log_2 N</td>
<td>1</td>
<td>20</td>
<td>-20</td>
</tr>
<tr>
<td></td>
<td>4D Tree</td>
<td>5</td>
<td>2Log_4 N</td>
<td>2Log_4 N - 2/3</td>
<td>10</td>
<td>9.33</td>
<td></td>
</tr>
<tr>
<td></td>
<td>kD</td>
<td>k+1</td>
<td>Log_k N</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2D fat tree</td>
<td>4</td>
<td>Log_2 N</td>
<td>N</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2D butterfly</td>
<td>4</td>
<td>Log_2 N</td>
<td>N/2</td>
<td>20</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>
Butterfly

Multistage: nodes at ends, switches in middle

- All paths equal length
- Unique path from any input to any output
- Conflicts that try to avoid
- Don’t want algorithm to have to know paths

Example MPP Networks

<table>
<thead>
<tr>
<th>Name</th>
<th>Number</th>
<th>Topology</th>
<th>Bits</th>
<th>Clock</th>
<th>Link</th>
<th>Bisect.</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>nCube/ten</td>
<td>1-1024</td>
<td>10-cube</td>
<td>1</td>
<td>10 MHz</td>
<td>1.2</td>
<td>640</td>
<td>1987</td>
</tr>
<tr>
<td>iPSC/2</td>
<td>16-128</td>
<td>7-cube</td>
<td>1</td>
<td>16 MHz</td>
<td>2</td>
<td>345</td>
<td>1988</td>
</tr>
<tr>
<td>MP-1216</td>
<td>32-512</td>
<td>2D grid</td>
<td>1</td>
<td>25 MHz</td>
<td>3</td>
<td>1,300</td>
<td>1989</td>
</tr>
<tr>
<td>Delta</td>
<td>540</td>
<td>2D grid</td>
<td>16</td>
<td>40 MHz</td>
<td>40</td>
<td>640</td>
<td>1991</td>
</tr>
<tr>
<td>CM-5</td>
<td>32-2048</td>
<td>fat tree</td>
<td>4</td>
<td>40 MHz</td>
<td>20</td>
<td>10,240</td>
<td>1991</td>
</tr>
<tr>
<td>CS-2</td>
<td>32-1024</td>
<td>fat tree</td>
<td>8</td>
<td>70 MHz</td>
<td>50</td>
<td>50,000</td>
<td>1992</td>
</tr>
<tr>
<td>Paragon</td>
<td>4-1024</td>
<td>2D grid</td>
<td>16</td>
<td>100 MHz</td>
<td>200</td>
<td>6,400</td>
<td>1992</td>
</tr>
<tr>
<td>T3D</td>
<td>16-1024</td>
<td>3D Torus</td>
<td>16</td>
<td>150 MHz</td>
<td>300</td>
<td>19,200</td>
<td>1993</td>
</tr>
</tbody>
</table>

MBytes/second

No standard MPP topology!
Connection-Based vs. Connectionless

- **Telephone:** operator sets up connection between the caller and the receiver
  - Once the connection is established, conversation can continue for hours

- Share transmission lines over long distances by using switches to multiplex several conversations on the same lines
  - “Time division multiplexing” divide B/W transmission line into a fixed number of slots, with each slot assigned to a conversation

- Problem: lines busy based on number of conversations, not amount of information sent
- Advantage: reserved bandwidth

**Connection-Based vs. Connectionless**

- **Connectionless:** every package of information must have an address => packets
  - Each package is routed to its destination by looking at its address
  - Analogy, the postal system (sending a letter)
  - also called “Statistical multiplexing”
Routing Messages

• Shared Media
  – Broadcast to everyone

• Switched Media needs real routing. Options:
  – Source-based routing: message specifies path to the destination (changes of direction)
  – Virtual Circuit: circuit established from source to destination, message picks the circuit to follow
  – Destination-based routing: message specifies destination, switch must pick the path
    » deterministic: always follow same path
    » adaptive: pick different paths to avoid congestion, failures
    » Randomized routing: pick between several good paths to balance network load

Deterministic Routing Examples

• mesh: dimension-order routing
  – \((x_1, y_1) \rightarrow (x_2, y_2)\)
  – first \(\Delta x = x_2 - x_1\)
  – then \(\Delta y = y_2 - y_1\),

• hypercube: edge-cube routing
  – \(X = x_0x_1x_2\ldots x_n \rightarrow Y = y_0y_1y_2\ldots y_n\)
  – \(R = X \text{xor} Y\)
  – Traverse dimensions of differing address in order

• tree: common ancestor
• Deadlock free?
Store and Forward vs. Cut-Through

- **Store-and-forward policy**: each switch waits for the full packet to arrive in switch before sending to the next switch (good for WAN)
- **Cut-through routing and worm hole routing**: switch examines the header, decides where to send the message, and then starts forwarding it immediately
  - In **worm hole routing**, when head of message is blocked, message stays strung out over the network, potentially blocking other messages (needs only buffer the piece of the packet that is sent between switches). CM-5 uses it, with each switch buffer being 4 bits per port.
  - **Cut through routing** lets the tail continue when head is blocked, accordioning the whole message into a single switch. (Requires a buffer large enough to hold the largest packet).

Store and Forward vs. Cut-Through

- **Advantage**
  - Latency reduces from function of:
    
    number of intermediate switches \( X \) by the size of the packet

    to

    time for 1st part of the packet to negotiate the switches
    + the packet size \( \div \) interconnect BW
**Congestion Control**

- Packet switched networks do not reserve bandwidth; this leads to **contention** (connection based limits input)
- Solution: prevent packets from entering until contention is reduced (e.g., freeway on-ramp metering lights)
- Options:
  - **Packet discarding**: If packet arrives at switch and no room in buffer, packet is discarded (e.g., UDP)
  - **Flow control**: between pairs of receivers and senders; use feedback to tell sender when allowed to send next packet
    - **Back-pressure**: separate wires to tell to stop
    - **Window**: give original sender right to send N packets before getting permission to send more; overlaps latency of interconnection with overhead to send & receive packet (e.g., TCP), adjustable window
  - **Choke packets**: aka “rate-based”; Each packet received by busy switch in warning state sent back to the source via choke packet. Source reduces traffic to that destination by a fixed % (e.g., ATM)

**Practical Issues for Interconnection Networks**

- **Standardization advantages:**
  - low cost (components used repeatedly)
  - stability (many suppliers to chose from)
- **Standardization disadvantages:**
  - Time for committees to agree
  - When to standardize?
    - Before anything built? => Committee does design?
    - Too early suppresses innovation
- **Perfect interconnect vs. Fault Tolerant?**
  - Will SW crash on single node prevent communication? (MPP typically assume perfect)
- **Reliability (vs. availability) of interconnect**
### Practical Issues

<table>
<thead>
<tr>
<th>Interconnection</th>
<th>MPP</th>
<th>LAN</th>
<th>WAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example</td>
<td>CM-5</td>
<td>Ethernet</td>
<td>ATM</td>
</tr>
<tr>
<td>Standard</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Fault Tolerance?</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Hot Insert?</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

- Standards: required for WAN, LAN!
- **Fault Tolerance**: Can nodes fail and still deliver messages to other nodes? required for WAN, LAN!
- **Hot Insert**: If the interconnection can survive a failure, can it also continue operation while a new node is added to the interconnection? required for WAN, LAN!

### Networks on a Chip
Message

- Interconnect-oriented architecture can reduce the demand for interconnect bandwidth and the effect of interconnect latency by an order of magnitude through
  - locality - eliminate *global* structures
  - hierarchy - expose locality in register architecture
  - networking - share the wires

Outline

- Technology constraints
- On-chip interconnection networks
  - regular wiring - well characterized
  - optimized circuits
  - efficient usage
- Placement and Migration
- Architecture for locality
On-chip wires are getting slower

- Reach in mm reduced to 0.35
- Reach in lambda reduced to 0.70

Technology scaling makes communication the scarce resource

<table>
<thead>
<tr>
<th>Year</th>
<th>Technology</th>
<th>DRAM Capacity</th>
<th>FP Proc Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1999</td>
<td>0.18µm</td>
<td>256Mb DRAM</td>
<td>16 64b FP Proc 500MHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>18mm</td>
<td>1 clock repeaters every 3mm</td>
</tr>
<tr>
<td>2008</td>
<td>0.07µm</td>
<td>4Gb DRAM</td>
<td>256 64b FP Proc 2.5GHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>25mm</td>
<td>16 clocks repeaters every 0.5mm</td>
</tr>
</tbody>
</table>
**On-Chip Interconnection Networks**

- Replace dedicated global wiring with a shared network

![Diagram of on-chip interconnection network](image)

**Most Wires are Idle Most of the Time**

- Don’t dedicate wires to signals, share wires across multiple signals
- Route packets not wires
- Organize global wiring as an on-chip interconnection network
  - allows the wiring resource to be shared keeping wires busy most of the time
  - allows a single global interconnect to be re-used on multiple designs
  - makes global wiring regular and highly optimized
Dedicated wires vs. Network

<table>
<thead>
<tr>
<th>Dedicated Wiring</th>
<th>On-Chip Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spaghetti wiring</td>
<td>Ordered wiring</td>
</tr>
<tr>
<td>Variation makes it hard to model</td>
<td>No variation, so easy to exactly</td>
</tr>
<tr>
<td>crosstalk, returns, length, R &amp; C.</td>
<td>model XT, returns, R and C.</td>
</tr>
<tr>
<td>Drivers sized for wire model’ – – 99% too large, 1% too small</td>
<td>Driver sized exactly for wire</td>
</tr>
<tr>
<td>Hard to use advanced signaling</td>
<td>Easy to use advanced signaling</td>
</tr>
<tr>
<td>Low duty factor</td>
<td>High duty factor</td>
</tr>
<tr>
<td>No protocol overhead</td>
<td>Small protocol overhead</td>
</tr>
</tbody>
</table>

On-Chip Interconnection Networks

- Many chips, same global wiring
  - carefully optimized wiring
  - well characterized
  - optimized circuits
    - 0.1x power 0.3x delay
- Efficient protocols
  - dynamic routing with pipelined control
  - statically scheduled
  - static
- Standard interface
Circuits for On-Chip Networks

Uniform, well characterized lines enable custom circuits - 0.1x power, 3x velocity

H-bridge driver
100mV swing

Long, lossy RC lines

Regenerative Repeaters

Interconnect: repeaters with switching

- Need repeaters every 1mm or less
- Easy to insert switching
  - zero-cost reconfiguration
- Minimize decision time
  - static routing
    - fixed or regular pattern
  - source routing
    - on-demand
    - requires arbitration and fanout
    - can be pipelined
- Minimize buffering
**Architecture for On-Chip Networks**

- **Topology** - different constraints than off-chip networks
  - buffering is expensive, bandwidth is cheap
  - more wires between ‘tiles’ than needed for one channel
    » multiple networks, higher dimensions, express channels

- **Flow-control**
  - flit-reservation flow control, pipeline control ahead of data
    » latency comparable to statically scheduled networks
    » minimum buffering requirements
  - run static, statically scheduled, and dynamic networks on one set of wires

- **Interface Design**
  - standard interface from modules to network
    » pinout and protocol
    » independent of network implementation

---

**Flit Interleave vs Virtual Channels**
(flow control through control layer) (6-flit message)

- Wormhole: 40bufs
- Wormhole: 80bufs
- Wormhole: 160bufs
- VirtualChannels: 2vcsX4bufs = 40bufs
- VirtualChannels: 4vcsX4bufs = 80bufs
- VirtualChannels: 8vcsX4bufs=160bufs
- VirtualChannels: 8vcsX8bufs=320bufs
- Interleave: 40bufs
- Interleave: 80bufs
- Interleave: 160bufs
**SIMD and Distributed Register Files**

### Organizations

- **48 ALUs (32-bit), 500 MHz**
- **Stream organization improves central organization by**
  - Area: 195x, Delay: 20x, Power: 430x
Performance

16% Performance Drop
(8% with latency constraints)

180x Improvement

Research Vehicle, The Chip of 2010

Integrated Processor-Memory Architecture (Stanford, MIT)
Coupled Osc. Clock Distribution (MIT)
Optical Clock Distribution (MIT)
Fast Drivers (Stanford)
Low-Power Drivers (MIT)
Short-Wire Arch. Georgia Tech
Partitioned Reg Organization (Stanford)
Data Migration (Stanford)
On-Chip Networks (Stanford)

L=0.07um, 25mm/side, 100K tracks/side
Architecture Reduces Impact of Slow Wires
Circuits Make Wires More Efficient

- **Locality**
  - Eliminate *implicit* global communication
  - Expose and optimize the communication
  - Clustered architecture
  - Partitioned register file
  - Data migration

- **Networking**
  - Route packets, not wires
  - Improves duty factor of wires
  - Single, regular, highly-optimized design

Protocols: HW/SW Interface

- **Internetworking**: allows computers on independent and incompatible networks to communicate reliably and efficiently;
  - Enabling technologies: SW standards that allow reliable communications without reliable networks
  - Hierarchy of SW layers, giving each layer responsibility for portion of overall communications task, called protocol families or protocol suites

- **Transmission Control Protocol/Internet Protocol (TCP/IP)**
  - This protocol family is the basis of the Internet
  - IP makes best effort to deliver; TCP guarantees delivery
  - TCP/IP used even when communicating locally: NFS uses IP even though communicating across homogeneous LAN
FTP From Stanford to Berkeley

- BARRNet is WAN for Bay Area
- T1 is 1.5 mbps leased line; T3 is 45 mbps; FDDI is 100 mbps LAN
- IP sets up connection, TCP sends file

Protocol

- Key to protocol families is that communication occurs logically at the same level of the protocol, called peer-to-peer, but is implemented via services at the lower level
- Danger is each level increases latency if implemented as hierarchy (e.g., multiple check sums)
**TCP/IP packet**

- Application sends message
- TCP breaks into 64KB segments, adds 20B header
- IP adds 20B header, sends to network
- If Ethernet, broken into 1500B packets with headers, trailers
- Header, trailers have length field, destination, window number, version, ...

**Example Networks**

- Ethernet: shared media 10 Mbit/s proposed in 1978, carrier sensing with exponential backoff on collision detection
- 15 years with no improvement; higher BW?
- Multiple Ethernets with devices to allow Ethernets to operate in parallel!
- 10 Mbit Ethernet successors?
  - FDDI: shared media (too late)
  - ATM (too late?)
  - Switched Ethernet
  - 100 Mbit Ethernet (Fast Ethernet)
  - Gigabit Ethernet
Connecting Networks

- **Bridges**: connect LANs together, passing traffic from one side to another depending on the addresses in the packet.
  - operate at the **Ethernet protocol level**
  - usually simpler and cheaper than routers
- **Routers or Gateways**: these devices connect LANs to WANs or WANs to WANs and resolve incompatible addressing.
  - Generally slower than bridges, they operate at the **internetworking protocol (IP) level**
  - Routers divide the interconnect into separate smaller subnets, which simplifies manageability and improves security
- **Cisco** is major supplier; basically special purpose computers

Networking Summary

- Protocols allow heterogeneous networking
- Protocols allow operation in the presence of failures
- Routing issues: store and forward vs. cut through, congestion, ...
- Standardization key for LAN, WAN
- Internetworking protocols used as LAN protocols => large overhead for LAN
- Integrated circuit revolutionizing networks as well as processors