

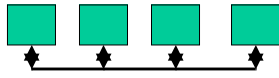
Communication Networks

Ian Watson & Mikel Lujan
Advanced Processor Technologies Group

The Need for Networks

- Any multi-core system must clearly contain the means for cores to communicate
 - With memory
 - With each other (probably)
- We have briefly discussed busses as the standard multi-core network
- Others are possible
 - But have different characteristics
 - May provide different functionality

Bus

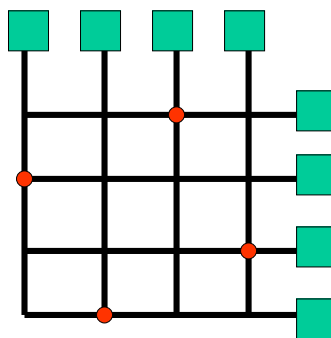


- Common wire interconnection
- Usually parallel wires - address + data
- Only single usage at any point in time
- Controlled by clock - divided into time slots
- Sender must 'grab' a slot (via arbitration)
- Then transmit (address + data)
- Often 'split transaction'
 - E.g send memory address in one slot
 - Data returned by memory in later slot
 - Intervening slots free for use by others

COMP60011 Future Multi-core Computing

Crossbar

- E.g to connect N cores to N memories

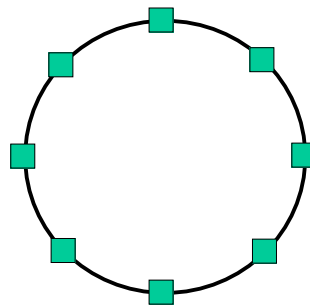


- Can achieve 'any to any' (disjoint) in parallel

COMP60011 Future Multi-core Computing

Ring

- Simple but
 - Low bandwidth
 - Variable delay



COMP60011 Future Multi-core Computing

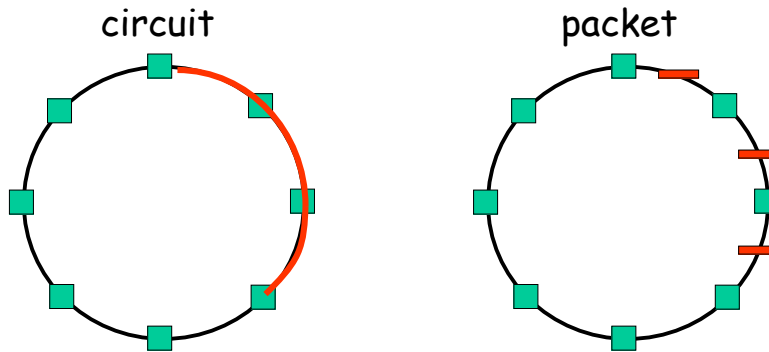
Packet Switched/Circuit Switched

- Circuit Switched
 - Direct physical connection - bus and crossbar are normally of this form
 - Means that when a communication is taking place it requires all of the resource i.e. physical wire
- Packet Switched
 - Data item carries address and is routed in stages
 - Normally involves buffering
 - Multiple transmissions can share the same medium

COMP60011 Future Multi-core Computing

Circuit / Packet

- Can be illustrated by ring



COMP60011 Future Multi-core Computing

Bandwidth vs. Latency

- Latency - time taken for communication
- Bandwidth - amount of data that can be sent per unit time
- Definitely not the same thing
 - A truck carrying one million 16Gbyte flash memory cards to London
 - Latency = 4 hours (14,000 secs)
 - Bandwidth = 8Tbit/sec ($8 * 10^{12}$ /sec)
 - A broadband internet connection
 - Latency = 100 microsec (10^{-4} sec)
 - Bandwidth = 10Mbit/sec ($10 * 10^6$ /sec)

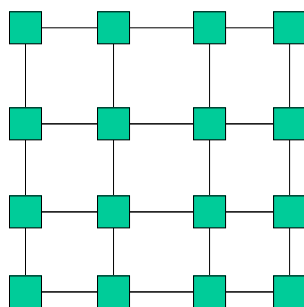
COMP60011 Future Multi-core Computing

Bandwidth vs. Latency

- Bandwidth vs Latency often a trade off
 - Can increase bandwidth at expense of longer latency
 - Can decrease latency at expense of lower bandwidth
- Packet switched often higher bandwidth but longer latency
- Ring a good example of this
 - Direct connection
 - Or multiple packets in transit

COMP60011 Future Multi-core Computing

Mesh / Grid



- Usually packet switched
- Reasonable bandwidth
- Variable Latency
- Convenient for VLSI physical layout
- 'wrap around' - becomes a toroid (ring doughnut)

COMP60011 Future Multi-core Computing

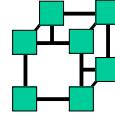
Hypercube



1D



2D

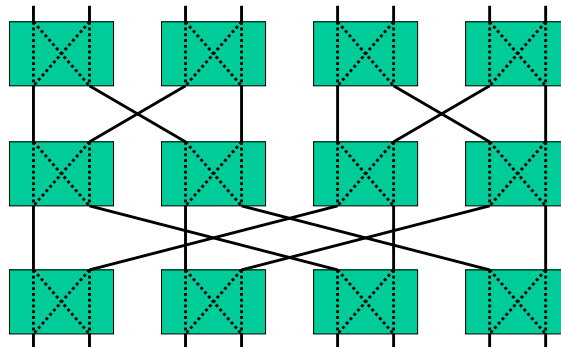


3D

nD?

- Each node has n connections to neighbours
- Will support 2^n nodes
- Max latency $O(n)$
- Usually packet switched
- Hard for VLSI layout (large n)

Multi-Stage (Banyan)



- Packet switched
- Many different detailed types - omega, flip etc.

Fault Tolerance

- In large systems (including future multi-core?) individual elements may fail
- Networks which have alternative routes to the same place (e.g. grid) can be made fault tolerant
- The more alternatives, the more fault tolerant
- However, overall complexity is increased
 - Usually need some form of acknowledgement (or maybe send redundancy) to make things work

Summary

- Lots of different choices (others not covered here)
- No obvious 'best' in most circumstances
- But - **important note** - MESI based cache coherence needs all cores to see all communication traffic - really only works with bus communication
 - Not scalable
 - Not fault tolerant