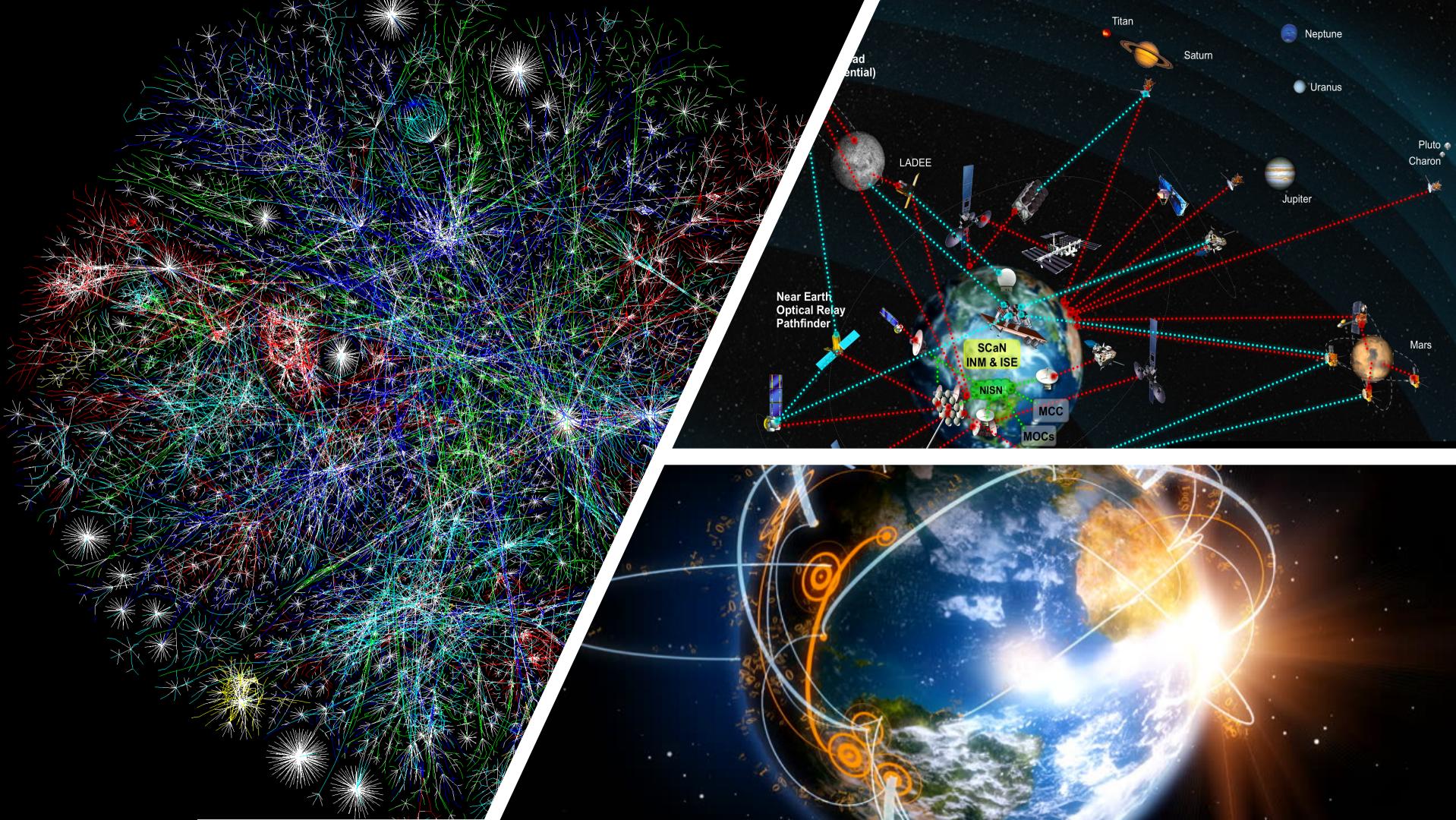
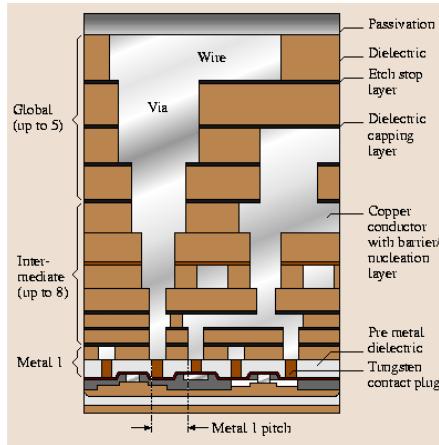


Network on Chip (NoC)

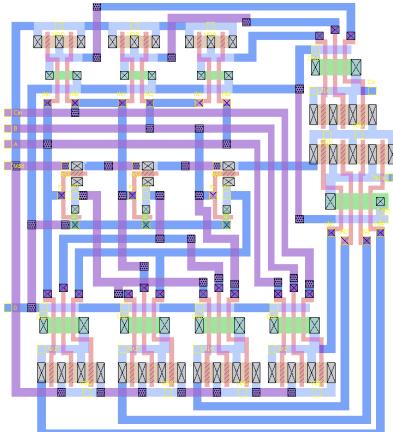
Professor Geoff Merrett *Based on slides from Dr Terrence Mak (UoS)*
ELEC3227: Embedded Networked Systems



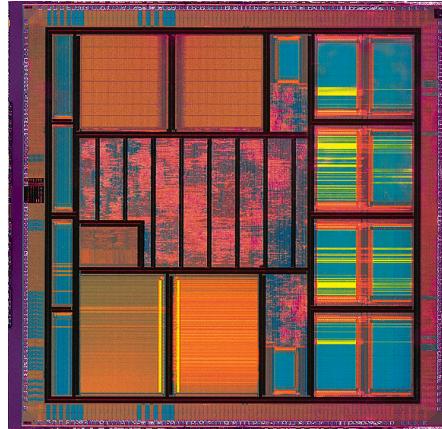
There are networks around the area of electronic computing



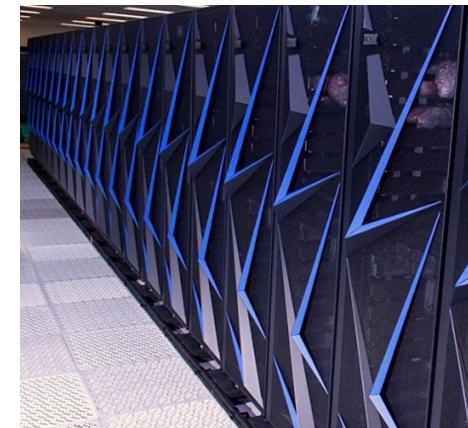
Vertical view of a chip with layers of metals and using via as connections



Layout of a multiplier using wires and silicon



Interconnects between components inside a very-large-scale-integration (VLSI)

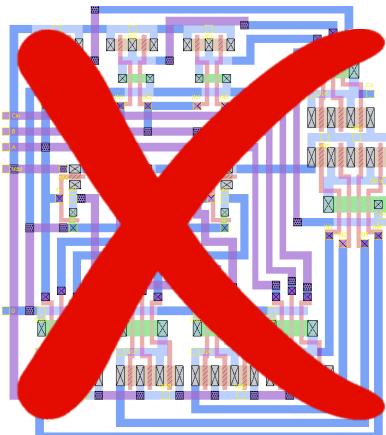


High-bandwidth interconnects for high-performance computing cluster

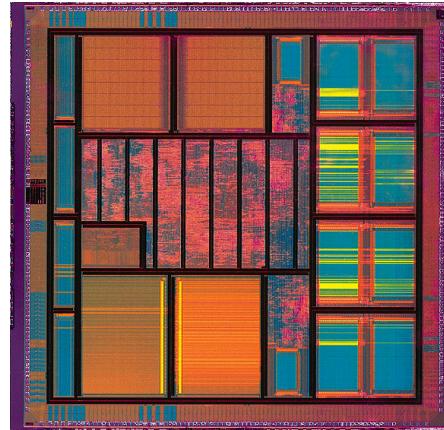
There are networks around the area of electronic computing



Vertical view of a chip with layers of metals and using via as connections



Layout of a multiplier using wires and silicon

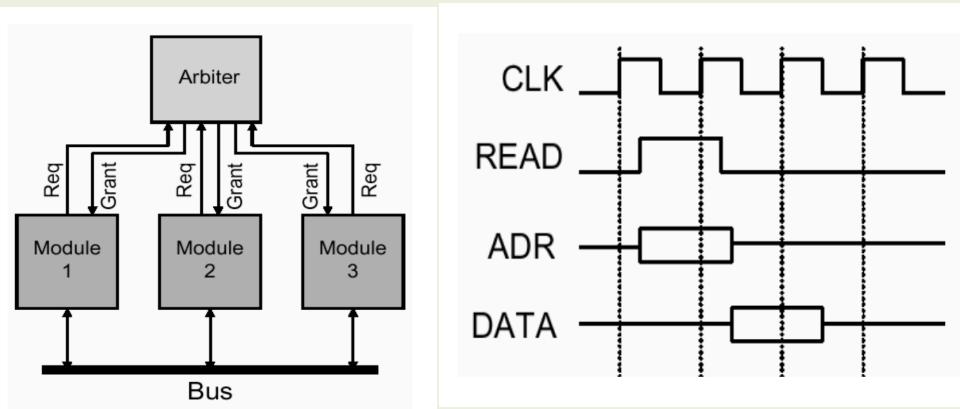


Interconnects between components inside a very-large-scale-integration (VLSI)

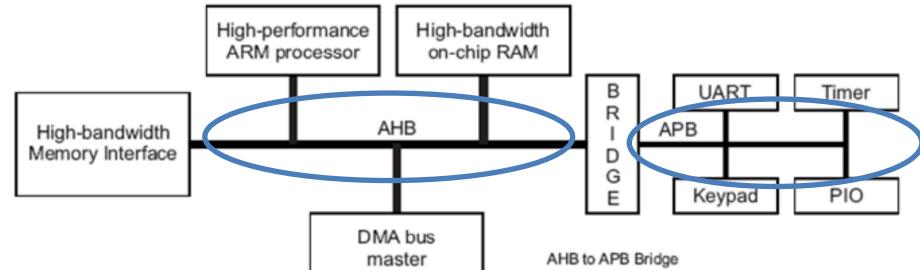


High-bandwidth interconnects for high-performance computing cluster

Classical Bus Architecture



- Bus provides one-to-one interconnect between modules
- Requires read/address as input, and data as output
- There are request and grant (based on the importance of the message)



- ARM processor using AHB (Advanced High-Performance Bus) and APB (Advanced Peripheral Bus)
- Pipelining of Address / Data
- Split Transactions
- Routing for multiple masters

Dr Terrence Mak

Challenges to Bus Architectures



The earliest system bus was invented in 1975 supporting Intel 8080; later AMBA was introduced by ARM in 1995



It used the design to provide connections between fewer components, eg. less than 10 components from ARM processor



Only provide one-to-one (or one-to-many) delivery



Limited bandwidth and parallelism



Appropriate for small (less number of components) or low-power devices

Networks On-Chip (NoC)

Characteristic of Network-on-Chip

- Well structured wiring, corresponding to the chip fabrication and management
- Different topologies available, suitable for different demand.
- Scalable, according to the needs of applications
- Short latency (different routes were provided), eg. depends on the requirement
- High-throughput, easily satisfies different on-chip components
- Provides multicasting or broadcasting
- Modular in nature, easy for verification

NETWORK ON CHIP

- N Computational Resources
 - Processing Elements (PEs)
- N Network Interfaces
- $M \leq N$ Switches
- 1 Connection Topology
- 1 Routing Technique

LAYERED COMMUNICATION

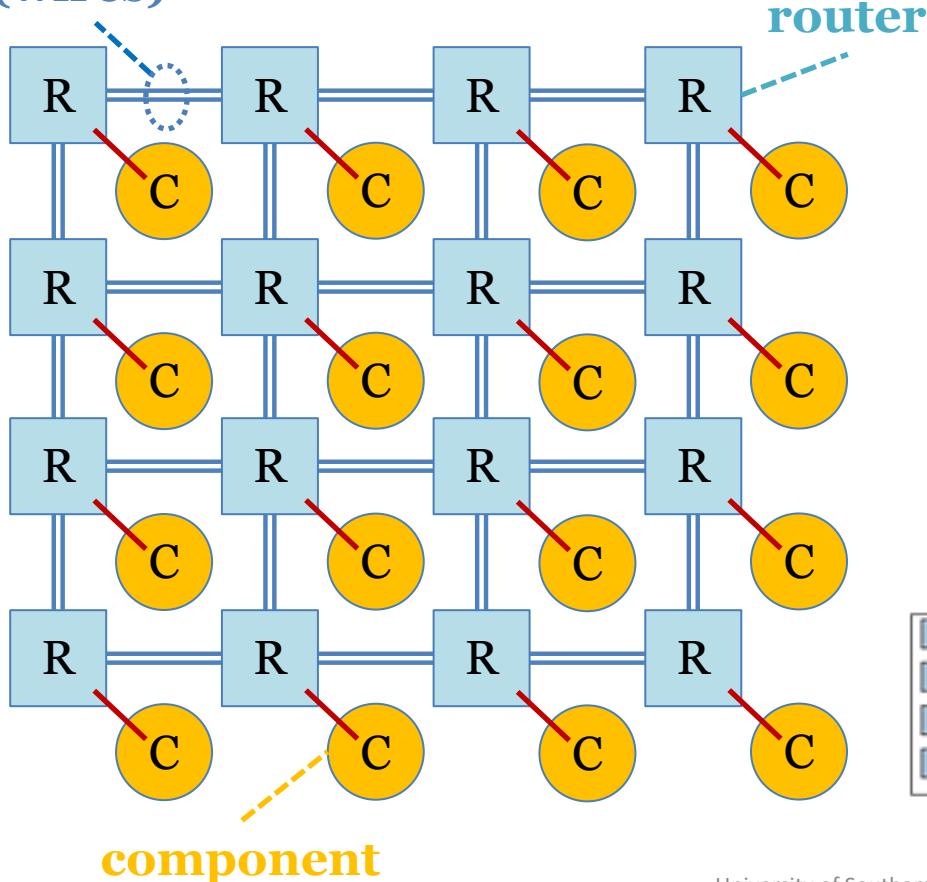
Units of Communication

- PHY: Word
- DLL: Flit (FLow control unIT)
- NET: Packet

Example Functions

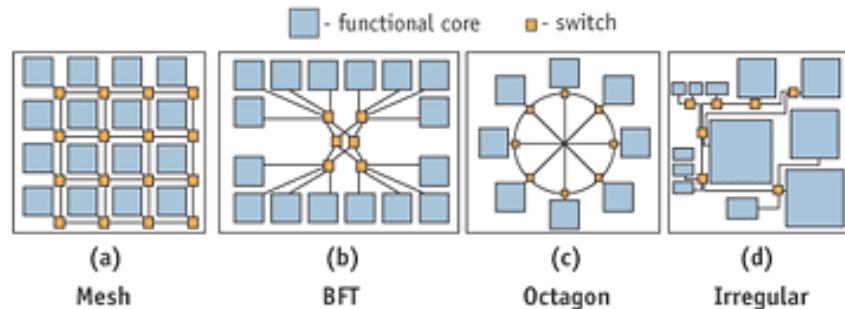
- DLL
 - Buffering
 - Error control
- NET
 - Routing
 - Addressing
 - Packets >> Flits
 - QoS (priorities etc)
 - Error control
- TRANS
 - Virtual channels

channel
(wires)



Network-on-Chip Architecture (Top Level)

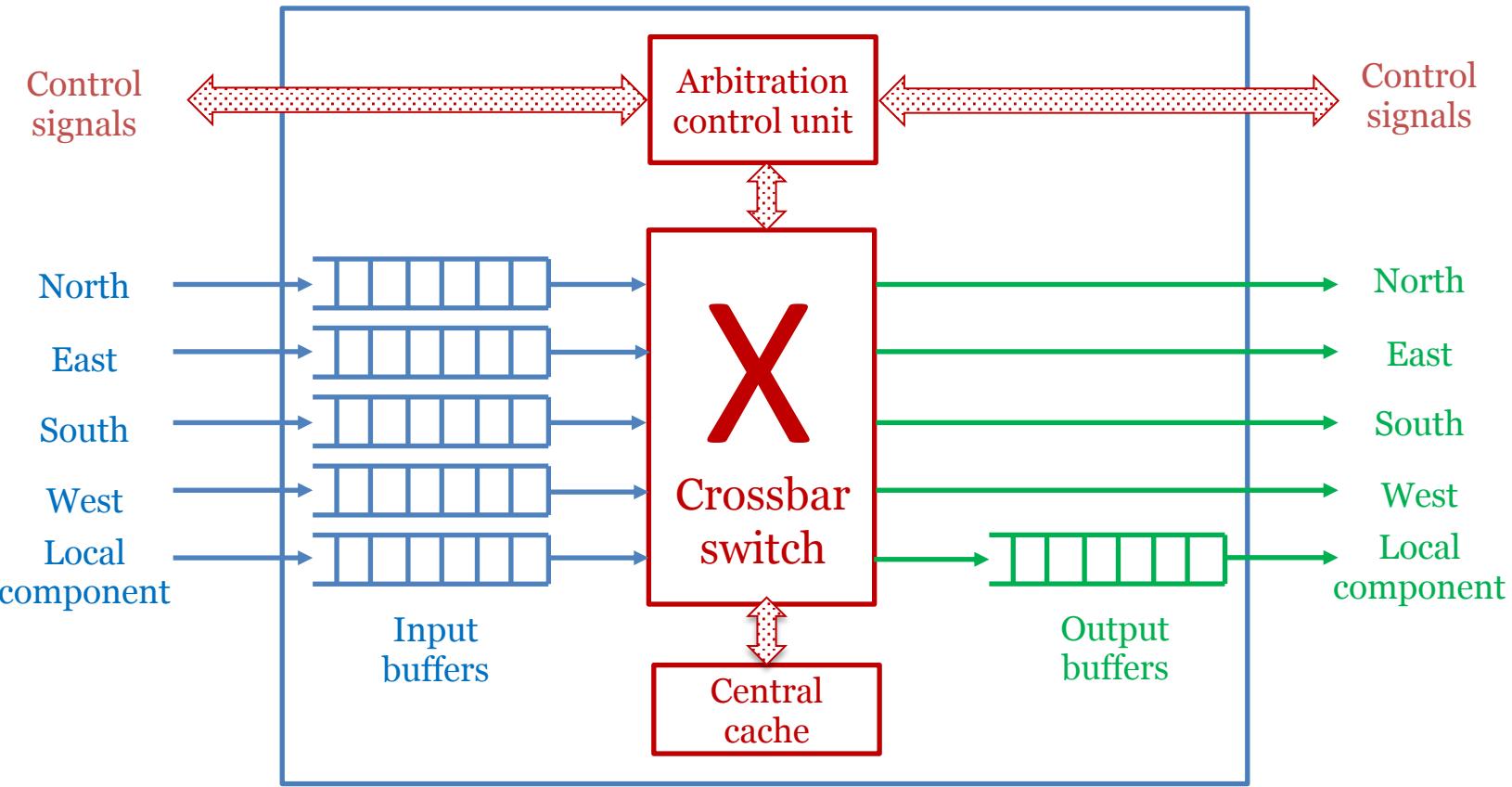
- Network-on-Chip consists of **routers**, **wires** and **components**.
- **Router** determined the directions to forward the packets
 - Connections between the router affects the overall traffics.
- **Wires** provide connections between routers
- **Components** can be varied
 - Cores, memory, sensor and power provider, etc.



SWITCHING

- Circuit switching
 - A physical path from source to destination is established, reserved, and released
 - Low complexity switches; Guaranteed QoS
- Packet switching (store and forward)
 - Entire packets are buffered at each switch
- Wormhole switching
 - First flit contains routing information, and tail follows (reduces buffering)

Network-on-Chip Architecture (Router Design)



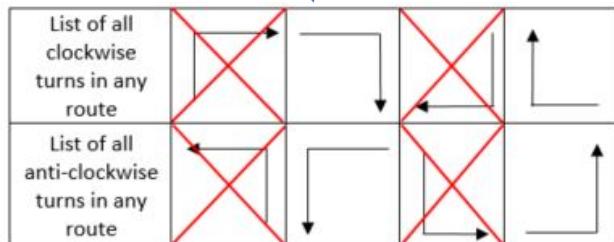
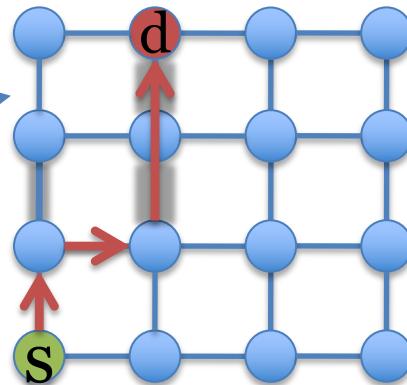
ROUTING

- Connection-oriented vs connectionless
- Minimal vs non-minimal routing
 - Only consider shortest paths vs allowing non-minimal routes
- Central vs decentralized
 - Global decisions vs local/distributed decisions
- Deterministic (oblivious¹) vs adaptive
 - Deterministic: predetermined; simple; often minimal
 - Adaptive: routes decided at runtime, based on network state (queues, historic load)

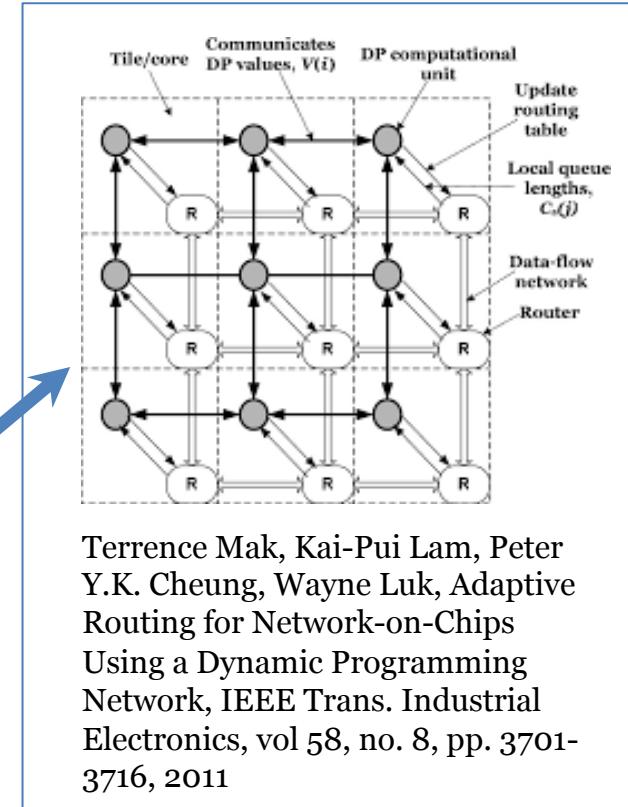
¹ *Oblivious = unaware of the network state; all random and deterministic algorithms*

What kind of routing strategy is best for you?

- Deterministic (Oblivious) routing
- XY routing



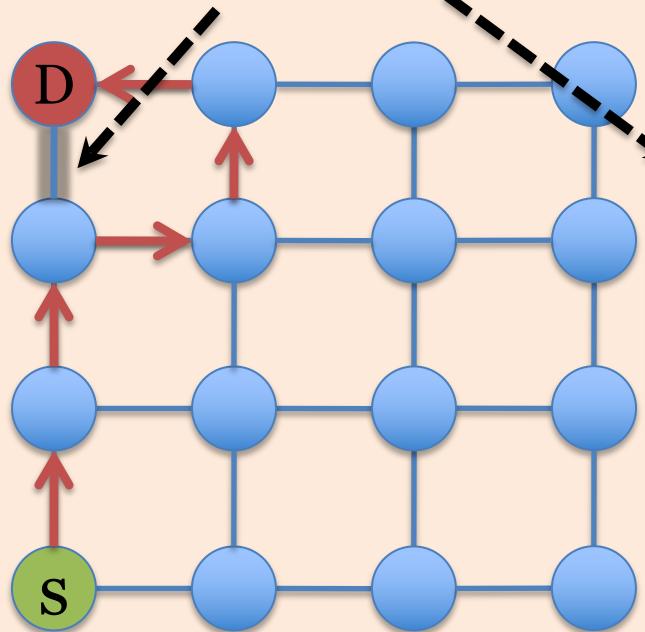
Adaptive routing -
based on the results of
run-time algorithm to
decide the next route



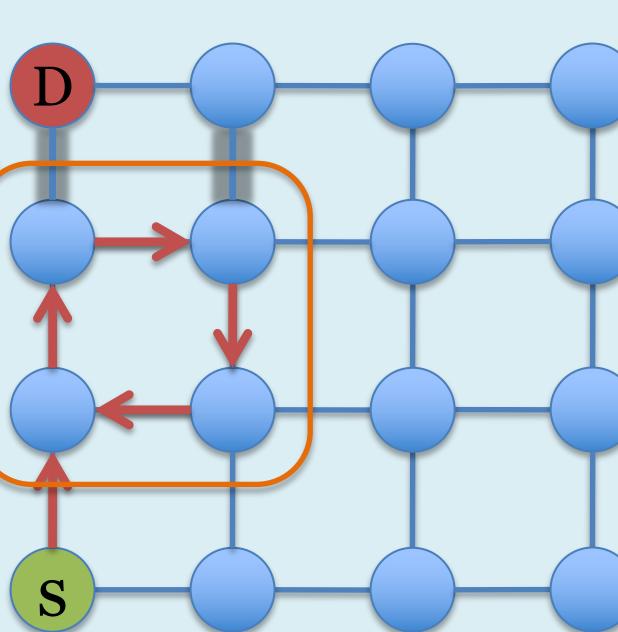
Terrence Mak, Kai-Pui Lam, Peter Y.K. Cheung, Wayne Luk, Adaptive Routing for Network-on-Chips Using a Dynamic Programming Network, IEEE Trans. Industrial Electronics, vol 58, no. 8, pp. 3701-3716, 2011

Fully adaptive routing and livelock

Intense traffic



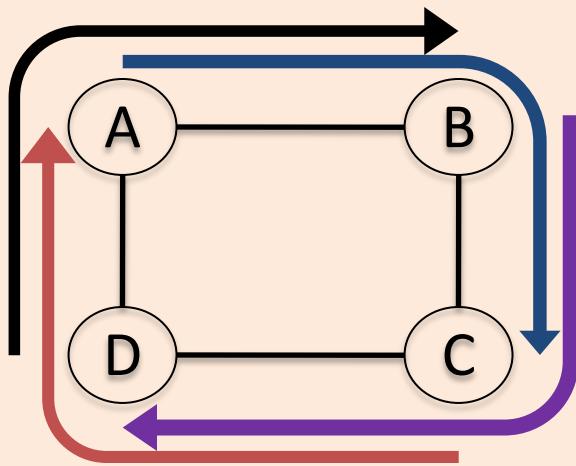
Longer path with potentially lower latency



Livelock: continue routing in cycle

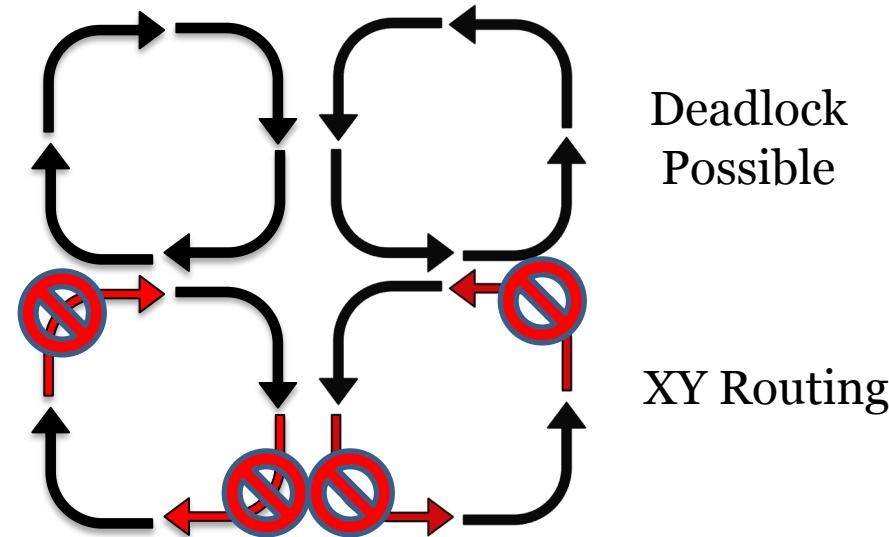
- Fully adaptive, not restricted to take shortest path (depends on the optimal definition)
- Misrouting: directing packet along non-productive channel
 - Priority given to productive output
 - Some algorithms forbid U-turns
- Livelock: traversing network without ever reaching destination
 - Mechanism to guarantee forward progress
 - misrouting

Routing Deadlock



Without routing restrictions,
a resource cycle can occur and
leads to **deadlock**

Eliminate Cycles by Construction



Don't allow turns that cause cycles.
There are turn models to avoid deadlock,
eg. west first, negative first etc.

Criteria which we concern



Switch Degree (Based on routing algorithm)



Latency (Impact of topology)



Throughput (Amount of data to transfer)



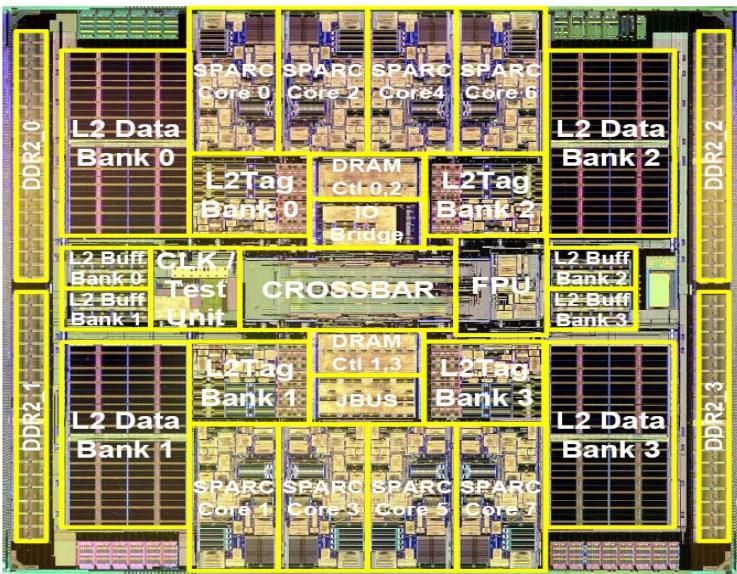
Maximum channel load (Max throughput)



Bisection Bandwidth (Overall bandwidth)



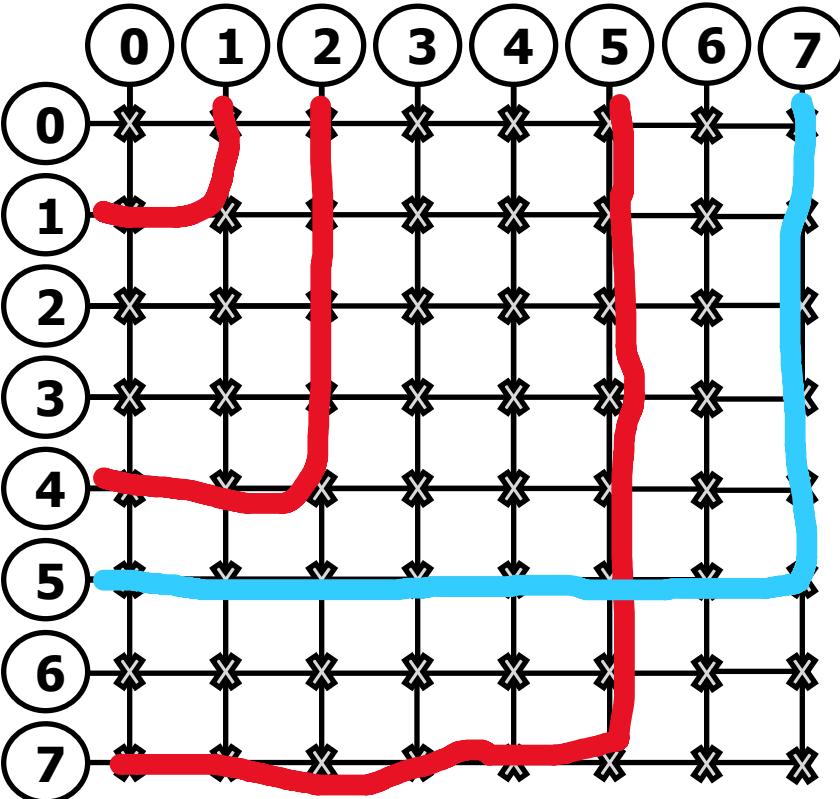
Path Diversity (Fault tolerance)



Sun Niagara

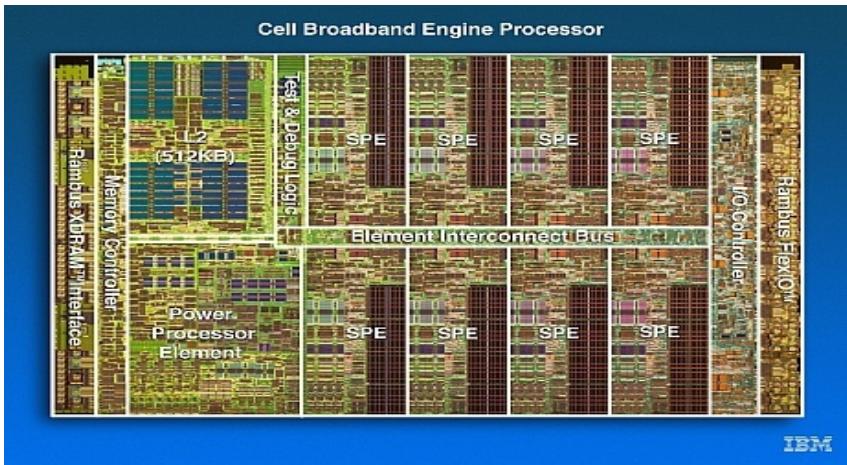
1. Maximize the throughput significantly
2. Easier to manage comparing to parallel Bus architecture
3. Scalable to have more core integrated

Multicore (1) - XBar



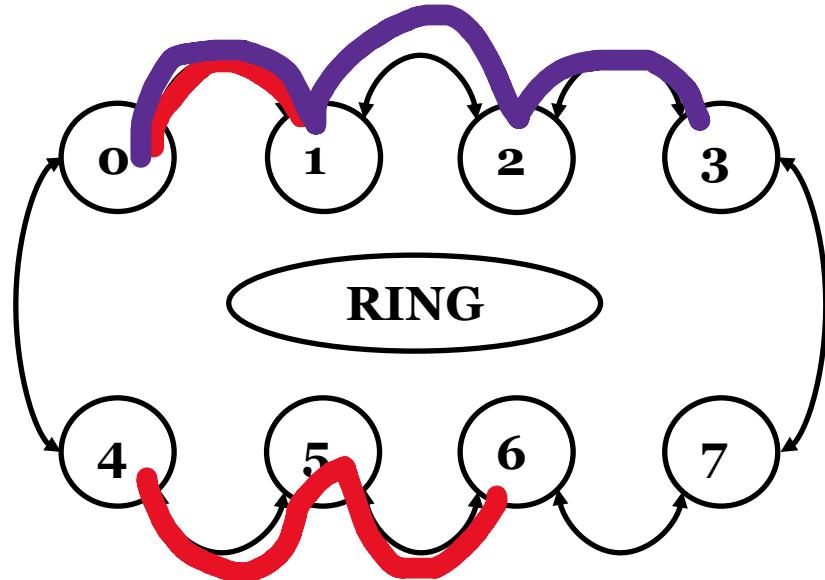
IBM Cell

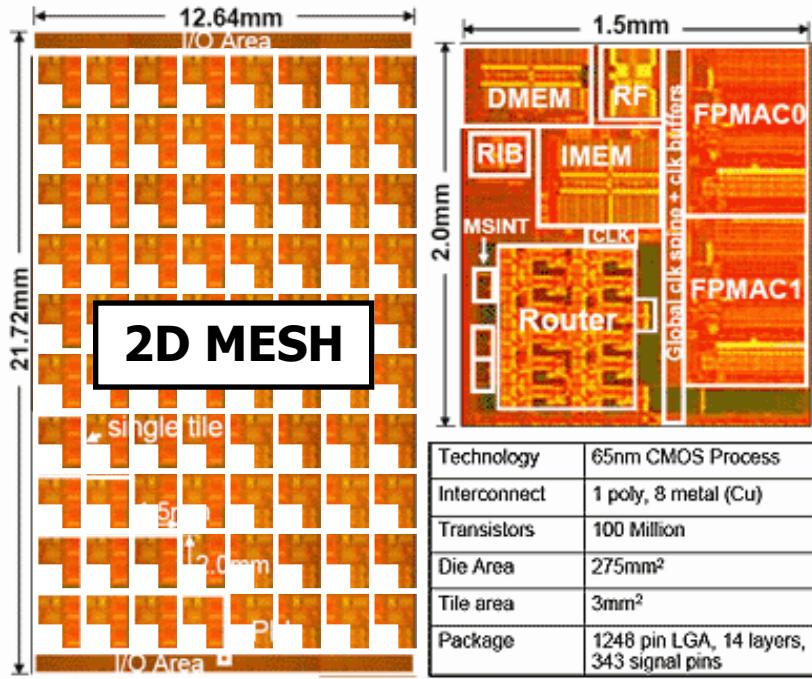
Multicore (2) - Ring



Element Interconnect Using Rings

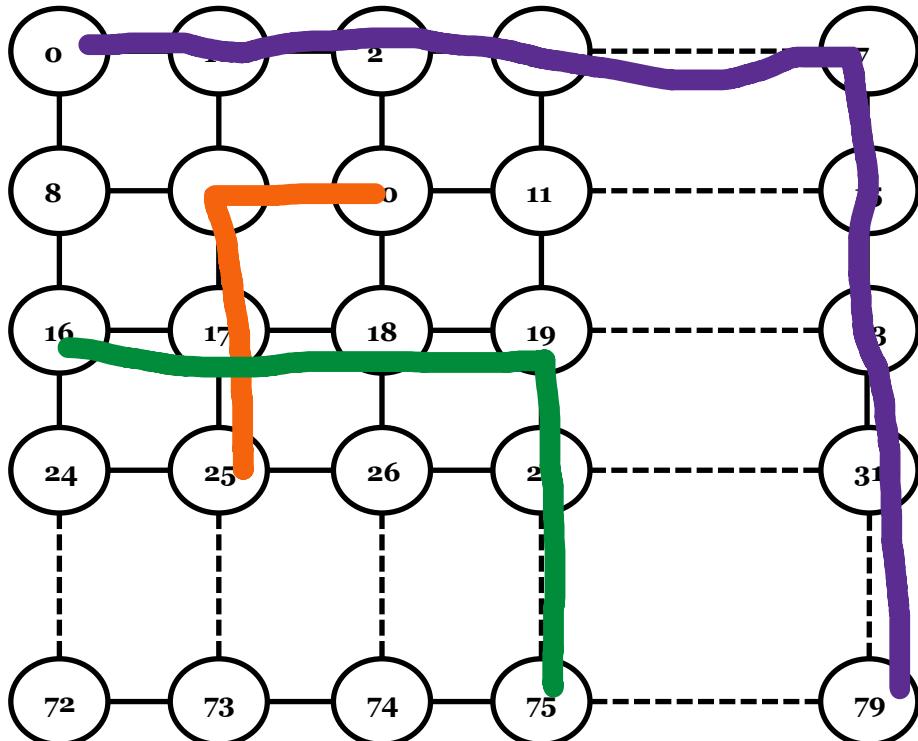
- Packet size: 16B-128B
- Credit-based flow control
- Up to 64 outstanding requests
- Latency: 1 cycle/hop





- Intel Polaris, with 80 core prototype
- Academic Research, eg. MIT Raw, TRIPs
 - 2-D Mesh Topology
 - Scalar Operand Networks

Many Core (3) – Mesh Topology



References

- Topology
 - William J. Dally and C. L Seitz. The torus routing chip. *Journal of Distributed Computing*, 1(3):187–196, 1986.
 - Charles Leiserson. Fat-trees: Universal networks for hardware efficient supercomputing. *IEEE Transactions on Computers*, 34(10), October 1985.
 - Boris Grot, Joel Hestness, Stephen W. Keckler, and OnurMutlu. Express cube topologies for on-chip networks. In *Proceedings of the International Symposium on High Performance Computer Architecture*, February 2009.
 - Flattened butterfly topology for on-chip networks. In *Proceedings of the 40th International Symposium on Microarchitecture*, December 2007.
 - J. Balfour and W. Dally. Design tradeoffs for tiled cmp on-chip networks. In *Proceedings of the International Conference on Supercomputing*, 2006.
- Routing
 - Terrence Mak, Kai-Pui Lam, Peter Y.K. Cheung, Wayne Luk, Adaptive Routing for Network-on-Chips Using a Dynamic Programming Network, *IEEE Trans. Industrial Electronics*, vol 58, no. 8, pp. 3701-3716, 2011
 - Raaed Al-Dujaily, Terrence Mak, Fei Xia, Alex Yakovlev and Maurizio Palesi, Embedded Transitive Closure Networks for Run-Time Deadlock Detection in Networks-on-Chip, *IEEE Trans. Parallel and Distributed Systems*, vol. 23, no. 7, pp. 1205-1215, 2012
 - L. G. Valiant and G. J. Brebner. Universal schemes for parallel communication. In *Proceedings of the 13th Annual ACM Symposium on Theory of Computing*, pages 263–277, 1981.
 - D. Seo, A. Ali, W.-T. Lim, N. Rafique, and M. Thottenhamdi. Near-optimal worst- case throughput routing in two dimensional mesh networks. In *Proceedings of the 32nd Annual International Symposium on Computer Architecture*, June.

UNIVERSITY OF
Southampton