On-Chip Networks for Multicore Systems

An OCN is defined by the topology, routing algorithm, flow control and router micro architecture. The topology of a network is how the nodes and links are connected and it is routed by the routing algorithm which picks the path a message can take through the network. The flow control is responsible for how a message traverses the route and the microarchitecture of a router realizes the flow and and routing control protocol for its circuits.

The different ways available to leverage OCN’s include having an explicit interface to modify the ISA, connect a network physically to the multicore system i.e. between FU’s, L1/L2 cache. All communication is done through units of packets. When a message is injected it is turned into a packet. 

In terms of topology, the topology of the network determines how long it takes to go from one part of the path to another, i.e. the number of hops required. In addition network reliability is also highly dependent on the topology as it also determines the reliability and what alternate paths there are to take. The most common are the mesh, torus, and ring. The ring topology provides few alternative paths, but have lower implementation overhead. A torus has a lower hop count, which means lees delay and energy there are more wire lengths folded to it increases the per hop latency and energy. All three are two dimensional planar topologies except for torus which needs to be set as folded wires.

In terms of routing algorithms, we need to find a way to evenly distribute the traffic among the paths. And is also important for fault tolerance in the system. The route that you pick directly affects the total energy. The most common algorithm that is used the DOR algorithm, Dimension-ordered-routing. Basically it travels dimension by dimension before reaching a coordinate matching the destination and then switching to the next dimension. [Picture: “DOR illustrates an X–Y route from (0,0) to (2,3) in a mesh, while Oblivious shows two alternative routes (X–Y and Y–X) between the same source–destination pair that can be chosen obliviously prior to message transmission. Adaptive shows a possible adaptive route that branches away from the X–Y route if congestion is encountered at(1,0)” (OCN, 40)]

One issue that occurs is deadlocks, in addition to energy, throughput, the reliability and must be free of deadlock. You can ensure deadlock freedom by preventing cycles. In terms of flow control, this is the allocation of the network buffer and link. It determines when buffers and links are assigned to which messages, the granularity at which they are allocated, and how these resources are shared among the many messages using the network (ocn, 42). Flow control also affects the network energy and power and the network quality of service because it is responsible for finding the arrival and departure time for packets at each hop. There is the store-and-forward protocol in which every node waits until it has seen a packet received before forwarding any part of the packet to another (next) node. This scheme has long delays at each hop and isn’t really good for OCNs. To reduce this part, there is also virtual cut through flow control which allows the packet to be transmitted before the packet is received. The tradeoff is
that although latency is reduced, the storage and bandwidth are allocated in packed sized units and only move if there is enough storage for it. Bad for OCNs with tight power and area constraints. The third way, is the *wormhole flow* which cuts through flits and lets them move to the next router when there is enough buffering for a flit (a packet is broken into flits), similar to virtual cut-through. *Virtual Channel Flow* improves on wormhole by allowing blocked packets to be passed by other packets, just consists of a separate flit queue in the router, and multiple VC’s share the physical links between two routers. Note that OCNs are not designed to tolerate dropping of packets, but have buffer backpressure to stall flit from being transmitted when buffer space is limited. They use it by having credits and on./off signaling. Credits = number of buffers and on/off isa signal between adjacent routers to indicate to send more or not.

In terms of router microarchitecture, it affects the path delay for critical paths, per hop delay, and overall network latency. It also affects the network energy as it determines how the circuit components in a router work and what their activity usage is. Major parts of the system are the input buffers, route computation logic, virtual channel allocator, switch allocator, and crossbar switch. Most OCNs are input buffered, i.e. packets are stored in buffers only at the input ports. To remove serialization during routing you can do head routing or pipeline bypassing which is used to shorten the router critical path, also speculation. Router buffers are built into SRAM. The switch design determines the ports and the allocators and arbiters are the VA allocators and must be fast and be able to be pipelined for high clock frequencies.

They case study two different networks. TRIPs is the interconnect between different FU’s on the system. And there is Intel’s TeraFlop which connects a lot more register files of 90 crores at a high rate. The router is each uses ports to connect to its neighbors. The teraflop is low power and scale pretty well.

The TRIPS process is a scalable architecture in which design is distributed and has scalable execution units and network integrated into the processing cores for ILP/TLP. Has a tiled architecture. the operand network delivers operands among the tiles. OPN architecture delivers operands among TRIPs with small latency and is similar to the RAW operand network.

The intel Chip has fine grained power management with the mesochronous communication for low power global clock distribution. Goal was to have a high performance low power short design cycle, by doing tiled and having c4 bumps, power and routing arrayed at the top level by combining them all. Complete turn around with traditional system in which all are fully customized designed processors. Uses hand optimized data path macros. and static gates for implementation speed.