Introduction

- Reading
  - for today Chapter 8
  - for Thursday Chapter 10
- Please email me the class you would like to lead discussion

Vector Timestamps in LRC

- Each Acquire/Release advances local clock
  - I.e. starts a new interval
- At Acquire:
  - compute new time at pair wise max of previous local and release.
  - Send old acquire timestamp to release node
  - Releasing node sends updates to node acquire node
- Maintain diffs for each node for each interval
- At Barrier, all participants in barrier see all changes
  - if barrier is global (all threads participate), can discard previous diffs

Interconnection Networks

- Performance Metrics
  - latency: time for the first bits to arrive
  - bandwidth: sustained volume of communication
  - connectivity: how far is the furthest neighbor
  - hardware cost: can we afford to build this thing?
  - reliability: can links fail and the system still run?
  - functionality: “for free” features
    - combining messages
    - fault tolerance
- Options
  - topology
  - operation mode (synchronous or asynchronous)
  - switching strategy (circuit, store-and-forward, cut-through)
  - control strategy (central or distributed)

Networks

- Static
  - Processors only communicate with neighbors
    - non-neighbor communication requires help of intermediate processors
    - this is a simple structure to build
    - performance suffers due to using processors to forward
  - Usual topologies are possible
    - hypercube, tree, mesh
- Dynamic
  - “switches” take care of forwarding between nodes
    - requires extra hardware to forward messages
    - hardware has to be able to route data
  - Topologies
    - bus, multi-stage, crossbar networks
    - any static topology

Switching Options

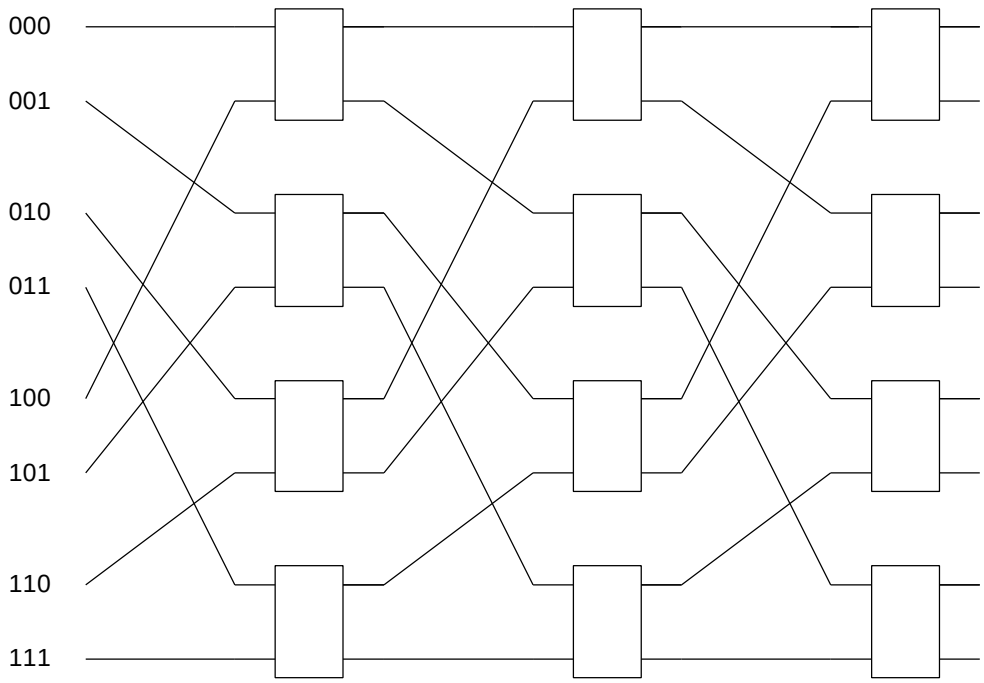
- Circuit switching
  - "connection" is established between end points
  - contention free connection is then used for data
  - latency is length + number of hops
- Store-and-forward
  - send hop by hop with the message being buffered
  - latency is message length * number of hops
- Cut-though (hybrid)
  - wormhole routing
    - first part of a message establishes a circuit
    - rest of message snakes its way long the path
  - possible to deadlock if not careful

Multistage Switching Networks

- several stages between source and destination
- provide different options in
  - cost
  - wires
  - blocking probability
- can be combined into many hybrid options

Omega Networks

- $\log_n(M)$ stages for M nodes and n by n cross bars
- message block if the required output link is not available



Benes networks

- additional stages to reduce blocking
- can route any disjoint src/dest pairs wo blocking
  - but must know requests in advance