Hardware-Based Routers

- Structure of hardware-based router:
  - Individual input and output ports with dedicated resources
  - Switch fabric as interconnect
  - Routing processor for control operations (e.g., routing)
### Input Ports

- **Processing on input port**
  - Layer 1:
    - Receive data
  - Layer 2:
    - Data link protocol
  - Layer 3:
    - IP address lookup
      - Copy of FIB
    - Forwarding
    - Queuing for switching fabric (NOT output queuing!)

- **Forwarding functions**
  - Check IP header checksum
  - Decrement TTL and adjust checksum

### Output Ports

- **Processing on output port**
  - Layer 3:
    - Buffering of packets
    - Scheduling decision which packet to send next
  - Layer 2:
    - Data link protocol
  - Layer 1:
    - Transmit data

- **Complete functionality specified in RFC 1812**
  - All functions that MUST, SHOULD, and MAY be implemented
Switching via Memory

- **Shared memory**
  - Memory interface for each input and each output port
  - Input port writes packet into queue of output port
- **Pros**
  - Simple design
  - Queue memory is shared among all ports
    - Reduces memory requirement
- **Cons**
  - Memory bandwidth at least 2NR (N=number of ports, R=link rate)
  - Memory speed grows at lower rate than link speed

Switching via Bus

- **Bus**
  - Single shared interconnect
  - One packet at a time
    - Bus needs to be fast
- **Pros**
  - Simple design
- **Cons**
  - Bus bandwidth needs to be NR
    - Does not scale due to capacitive loading
  - Requires arbitration if bus bandwidth less than NR
Switching via Crossbar

- **Crossbar**
  - Multiple interconnects that can be configured for each transmission
  - Crossbar controller determines connections
- **Pros**
  - Each connection bandwidth only \( R \)
- **Cons**
  - Requires algorithm to determine configuration
    - Optimal configuration might be difficult to determine quickly

Switching via Multistage Fabrics

- **Several stages of simple switching elements**
  - Switching element
    - Single control bit
  - Delta network
    - Recursive construction
  - Omega network
    - Uniform connections
- **Pros**
  - Each link requires only \( R \) bandwidth
  - Simple switching elements
  - Self-routing property
- **Cons**
  - Multiple stages
  - More complex designs for non-blocking
Output Queuing

- Where should we buffer packets?
- Output queued switch:
  - Inputs send to packets to output
    » Multiple packets may arrive in one cycle
  - Output buffers packets
    » Scheduler decides which to send next
- What’s the catch?
  - Worst case:
    » Port needs NR bandwidth
    » Aggregate bandwidth becomes N^R

Input Queuing

- Input queued switch:
  - Input queues packets
    » Only one packet per cycle is sent to any given output
    » Requires central coordination
  - Output can send packet right out
    » No buffering required
- What’s the catch?
  - Head-of-line blocking
  - Maximum throughput \((2-\sqrt{2})=0.586\) for large \(N\)
Virtual Output Queuing

- How can we avoid HOL blocking?
  - On input port, maintain one queue for each output port

Control algorithm necessary
- Problem can be modeled as bipartite graph
  - Line width indicates edge weight (# of packets)
- “Matching” determines conflict-free transmissions
  - At most one edge from any input
  - At most one edge to any output
  - “Maximum matching” allows most transmissions
iSLIP Algorithm

- Iterative matching algorithm [McKeown, ToN 1999]
- Three steps run repeatedly:
  1. Request: Each input sends a request to every output for which it has a queued cell
  2. Grant: If an output receives any requests, it chooses the one that appears next in a fixed, round-robin schedule of the inputs starting from the highest-priority input. The output notifies each input whether or not its request was granted.
  3. Accept: If an input receives a grant, it accepts the one that appears next in a fixed, round-robin schedule starting from the highest-priority output. The pointer to the highest-priority output is incremented (modulo $N$) to one location beyond the accepted output. Likewise, the pointer to the highest-priority input is incremented (modulo $N$) to one location beyond the granted input. The pointers are updated only after the first iteration; subsequent iterations match inputs and outputs that were not matched during earlier iterations.

Comparison of Switch Farbics

- From Business Communication Review 12/1997
- Assumptions
  - Line rate of 2.4 Gbps
  - 16 ports
  - Bus with 4x speedup
Fast-Path vs. Slow-Path

- Router is fine-tuned for fast handling of packets
  - Optimized for common case ("fast path")
  - Some packets require special processing
    » Processed on control processor ("slow path")

- "Unusual" packets
  - IP options
  - IP error handling (ICMP)
  - Packets addressed to router (routing)

Homework

- Read
  - Kurose & Ross: Chapter 4.7 (flooding and spanning tree)
- SPARK
  - Assessment quiz