Switching Techniques

Switching

• Determines *when* and *how* packets/messages are forwarded through the network

• Specifies the granularity and timing of packet progress

• Relationship with flow control has a major impact on performance
  - For example, overlapping switching and flow control
  - Overlapping routing and flow control

Concepts: Reminder

Basic Physical Channel Router

• Note the basic tension between performance and reliability
  - Degree of overlap between flow control, routing, and switching
Circuit Switching (1)

- Two phase protocols
  - Path setup + data transfer

- Buffers for "request" tokens

Circuit Switching (2)

- Two phase protocols
  - Path setup + data transfer

- Buffers for "request" tokens

- Request for circuit establishment
  (routing and arbitration is performed during this step)

Circuit Switching (3)

- Two phase protocols
  - Path setup + data transfer

- Buffers for "ack" tokens

- Acknowledgment and circuit establishment
  (as token travels back to the source, connections are established)

Circuit Switching (4)

- Two phase protocols
  - Path setup + data transfer

- Packet transport
  (neither routing nor arbitration is required)
Circuit Switching (5)

- Two phase protocols
  - Path setup + data transfer

High contention, low utilization → low throughput

Circuit Switching Example

- No message blocking once a circuit is setup.
- Secondary blocking due to setup
- Flow control is at the message level

Circuit Switching

- Hardware path setup by a routing header or probe
- End-to-end acknowledgment initiates transfer at full hardware bandwidth
- System is limited by signaling rate along the circuits
- Routing, arbitration and switching overhead experienced once/message

Circuit Switching Performance

\[ t_{circuit} = t_{setup} + t_{data} \]
\[ t_{setup} = D \left( t_{RC} + 2(t_{IB} + t_{ST} + t_{LT}) \right) \]
\[ t_{data} = \frac{1}{B} \left[ \frac{L}{W} \right] \]

- L = number of bits/packet
- D = number of hops to the destination
- W = channel width in bits
- B = channel bandwidth in hertz
Packet Switching (1)

- Switching
  - Store-and-forward switching

Buffers for data packets

Packets are completely stored before any portion is forwarded

---

Packet Switching (2)

- Switching
  - Store-and-forward switching

Requirement: buffers must be sized to hold entire packet (MTU)

Packets are completely stored before any portion is forwarded

---

Packet Switching (Store & Forward)

- Finer grained sharing of the link bandwidth
- Routing, arbitration, switching overheads experienced for each packet
- Increased storage requirements at the nodes

---

Packet Switching (Store & Forward)

- Packetization and in-order delivery requirements
  - Header overhead
- Alternative buffering schemes
  - Use of local processor memory
  - Central (to the switch) queues
Packet Switching: Example

- Packets block occupying buffers in a node
- Packets must be received in their entirety before being routed
- Flow control is at the packet level
- Reliable transmission can be handled at the link level

Packet Switching Performance

\[ t_{\text{packet}} = D \left\{ t_{RC} + (t_{IB} + t_{ST} + t_{LT}) \right\} \left\{ \frac{L + W}{W} \right\} \]

- L = number of bits/packet
- D = number of hops to the destination
- W = channel width in bits
- B = channel bandwidth in hertz

Virtual Cut-Through Switching (1)

- Switching
  - Cut-through switching

  [Diagram showing virtual cut-through switching]

  Portions of a packet may be forwarded ("cut-through") to the next switch before the entire packet is stored at the current switch

Virtual Cut-Through

- Messages can cut-through to the next router
- In the absence of blocking, messages are fully pipelined
  - Pipeline cycle time is determined by the router pipeline
- Routing and flow control is overlapped with transmission

[Diagram showing virtual cut-through]
Virtual Cut-Through

- When the header is blocked, the complete message is buffered at a switch
- High load behavior approaches that of packet switching

VCT Switching Example

- Packets block occupying buffers in a node
- Flow control is at the packet level
- Reliable transmission can be handled at the link level

VCT Performance

\[ t_{\text{VCT}} = D(t_{IB} + t_{RC} + t_{ST} + t_{LT}) + t_{IB} \left[ \frac{L}{W} \right] \]

- L = number of bits/packet
- D = number of hops to the destination
- W = channel width in bits
- B = channel bandwidth in hertz

Routing, Arbitration, and Switching

- Switching
  - Virtual cut-through
  - Wormhole

Buffers for data packets
Requirement: buffers must be sized to hold entire packet (MTU)
Buffers for flits: packets can be larger than buffers

Wormhole Switching

- Switching
  - Virtual cut-through
  - Wormhole

Buffers for data packets:
Requirement: buffers must be sized to hold entire packet (MTU)

Buffers for flits:
packets can be larger than buffers

Packet completely stored at the switch
Packet stored along the path

Source end node → Destination end node


Wormhole Switching: Example

- Messages cannot be interleaved over a channel: routing information is only associated with the header
- Packets block holding resources across multiple channels
- Flow control is at the flit level

Wormhole Performance

\[
t_{\text{wormhole}} = D(t_{IB} + t_{RC} + t_{ST} + t_{LT}) + t_{IB} \left( \frac{L}{W} \right)
\]

- L = number of bits/packet
- D = number of hops to the destination
- W = channel width in bits
- B = channel bandwidth in hertz

Small buffer extension to virtual cut through
Small buffers + message pipelining → small compact switches/routers
Supports variable sized messages

Does not include injection times
Impact of Virtual Channel Flow Control

- As the number of virtual channels increase, the increased channel multiplexing has two effects
  - Decrease in header delay
  - Increase in average data flit delay
- Both upstream and downstream link utilizations of this message will decrease

Impact of VCs on Routers

- Input queuing vs. output queuing
  - decoupling buffers@output
- Virtual channel state
  - Sender vs. receiver
- Switching options - #ports
  - Physical vs. virtual vs. hybrid crossbars
- VC allocation/de-allocation
  - When and who

Resource Allocation

- Two level resource allocation
  - VCs assigned on a packet basis
  - Channel bandwidth assigned on a flit basis
- VC state information
- On-chip impact →
  - buffer area vs. performance?
  - Wires (larger switches) vs. buffers?

Full Size VC Router

Input-Output Buffered Switch

Unidirectional Physical Channel

Per VC state

Virtual channel state

Switching options - #ports

Physical vs. virtual vs. hybrid crossbars

VC allocation/de-allocation

When and who
Properties of Switching Techniques

- Packet switching and virtual cut-through
  - Consume network bandwidth proportional to network load
  - Predictable demands
  - VCT behaves like wormhole at low loads and like packet switching at high loads
  - Link level error control for packet switching

- Wormhole switching
  - Provides low (unloaded) latency
  - Lower saturation point
  - Higher variance of message latency than packet or VCT switching
Comparison of Switching Techniques

- Reliability
  - Compatibility with traditional routing schemes
  - Effect of faults/failures

- Router implementation
  - Speed, power, area

- Saturation (blocking) load behavior

- Now for some switching variants

Mad Postman Switching

- Extend virtual cut-through to finer grained level
  - Speculate on the routing destination, e.g., maintain dimension in meshes
  - Make the common case fast

- Dead flits
  - Easily removed when blocked, e.g., at high loads

- Better match for low dimensional networks

Pipelined Circuit Switching

- Combines circuit switching and wormhole switching

- Paths are setup using virtual channels: reserved channels do not consume physical channel bandwidth

- Increased flexibility in routing the header flit

- Motivated by reliability concerns

- The use of multiple flit types lead to the use of control channels
Bufferless Switching

- Routers do not have buffers
- Each cycle, a packet must be directed to an output port
- The injection port is special
  - Inject a packet only when at least one input port is free
- Arbitration in the routers will ensure livelock freedom
- Packets may be misrouted

Bufferless Switching: Properties

- Simple router design
  - No flow control, no buffers
- Out of order packet delivery
  - Reordering at the destination is required for in-order packet delivery
- Bufferless networks saturate sooner
- Resolving collisions is key to performance
  - Need simple, fast arbitration

Switching: Global View

- Topology
  - Switching techniques are becoming less sensitive to topology being largely determined by wire delays
- Reliability
  - Switching technique will determine recovery schemes
- Deadlock Freedom
  - Choice typically depends on switching technique (we will study in detail)
- QoS
  - Global guarantees on “which” and “when” resources will be available?
- Switch architecture

What is Happening On-Chip?

- Bandwidth is plentiful, latency is in shorter supply
  - Wires are cheap & buffers are expensive
- Multiple clock cycles to cross long links
  - Repeater insertion
  - Latch insertion and pipelining wires
- Device variation, energy, and thermal effects (reliability?)
- How does this impact FC and switching?
Some Research Questions

- Quality of Service Guarantees
  - On bandwidth as well as latency

- Reliable communication
  - Link level error recovery
  - End-to-end error recovery

- Relationship between topology and switching
  - Reducing the network diameter for bufferless switching

- Handling asymmetry
  - How do you handle traffic asymmetry with multiple memory controllers?
  - Bandwidth allocation and switching

Summary

- Switching techniques interact with flow control to determine concurrency in router operation
- Deadlock and livelock freedom properties of routing protocols depend on switching techniques
- Choice of switching technique is a function of traffic patterns, offered load, and energy-performance budget