Interconnection Network Basics

Topics:

- How PEs send and receive messages.
- Buses
- Crossbars
- Direct Networks
- Multistage Networks

Terminology will be introduced as the network types are described.

How PEs Send Messages

Communication Paradigms

- *Message Passing*

  Programmer specifies what data to send and where to send it.

- *Shared Memory*

  Programs read and write a common *address space.*

  Programs run as though there were a single memory. . . .

  . . . in fact, memory is spread among the PEs. . .

  . . . with hardware sending messages between PEs to implement this.

Any network can support either paradigm.

In class the message passing paradigm will be primarily used.

Typical Procedure to Send a Message

1: Task calls message-passing function.

Function call specifies data, destination, and other data.

2: Message-passing function prepares a *message*.

Message consists of

the data to be sent,

the *destination*,

data needed by the network to route the network,

and other information.

3: A *request* [to send the message] is offered to the network through a *network interface*.

4: The message is copied into the network, en route to its destination.

Typical Procedure to Receive a Message

1: Network interface copies message to local memory.

2: Message may specify:

- A *handler* task which is to be started.

  In this case, the destination PE was not "expecting" the message.

- A task which is waiting for the message.

  Some task on the destination PE called some wait-for-message function.

Bus Networks

Structure

Consists of a single communication medium, the bus.

All PEs can read data from the bus.

At any time, at most one PE can place data on the bus.

An *arbitration mechanism* manages access to the bus.

Operation of Arbitration Mechanism

PEs issue requests to bus.

(Requests can be issued at any time.)

Arbitration mechanism chooses a request to *grant*.

Message corresponding to granted request is transfered, possibly in multiple steps..

Process is repeated after message transfer is complete.

Performance Measures—Amount of Data

Arrival Rate: rate at which requests are offered to the network, per input.

Typical unit: bytes per cycle.

Example: Let each PE send a message of 50 bytes every 100 cycles. Then the arrival rate is 0.5 bytes/cycle.

Saturating Traffic: a system in which there is always a request waiting at every network input.

Throughput: the amount of data a network can pass per input per unit time.

Typical unit: bytes per cycle per input.

Example: Each input of a network is offered requests at a rate of 1 request per 100 cycles. Each request is 20 bytes. All messages are delivered. Then the throughput is 0.2 (bytes/input)/cycle.

The two terms are similar.

Uses of Throughput Measure

Throughput refers to some test conditions.

- Running a benchmark (test) program.
  The network has a throughput of 1 GB/s running the radix sort program.

- A network offered saturating traffic.
  The network has a maximum throughput of 2.3 GB/s.

For parallel computers, throughput and arrival rate usually depend on each other.

Analysis of Bus

Determine throughput of bus offered saturating traffic.

Given:

$N = 64$ PEs connected to bus.

Bus width: $w = 8$ bytes.

Time to send w bytes of data: $t_x = 1$ cycle / transfer.

Time between transfers: $t_a = 1$ cycle.

Message size: $M = 256$ bytes.

Throughput, by definition, data per time per input.

Consider an interval in which one message sent.

Throughput is $\frac{M}{(M/w)t_x + t_a}\frac{1}{N} = \frac{4}{33} \approx 0.1212$ bytes / cycle.

For $N = 256$, throughput is 0.0303 bytes / cycle.

Performance Limits

Performance drops with number of processors.

Because of bus length, clock speed cannot easily be increased.

Because of cost, bus width cannot easily be increased.

Because of these limits, bus systems cannot have a large ($\gtrapprox 30$) number of processors.

Solutions

Multiple buses can be used.

Other network types can be used.

Crossbar Switch

Structure

An N-input, M-output crossbar consists of $N M$ switches, called *crosspoints*.

Each crosspoint can connect an input to an output.

An arbitration mechanism ensures that each output connects to no more than one input.

Operation

Request is offered to crossbar.

If destination is not currently receiving a message, message is sent.

Otherwise, request waits until destination idle.

Differences With Bus

Consider an N-input, N-output crossbar.

As many as N messages can be simultaneously en route. (Good.)

N^2 crosspoints needed. (Bad.)

Crossbar Throughput Analysis

Determine throughput of bus offered **ideal** saturating traffic.

Traffic is ideal if exactly one message is directed at each output.

Given:

$N = 64$ PEs connected to crossbar.

Datapath width: $w = 8$ bytes.

Time to send data: $t_x = 1$ cycle.

Time between transfers: $t_a = 1$ cycle.

Message size: $M = 256$ bytes.

Requests will only be made to idle outputs.

Throughput, by definition, data per time per input.

Consider an interval in which one message sent.

Throughput is $\dfrac{M}{(M/w)t_x + t_a} = \dfrac{256}{33} \approx 7.76$ bytes / cycle.

For $N = 256$, throughput is $\frac{256}{33} \approx 7.76$ bytes / cycle.

Note that throughput is independent of size.

Throughput Under Non-Ideal Conditions

Worst Case: all requests to same output.

Performance same as bus.

Blame the programmer. (Even if the network could handle the traffic, could the PE?).

Popular (but not necessarily realistic) Case:

Randomly chosen destinations.

For a two-input crossbar throughput is $\frac{3}{4}$ of ideal case.

For an N-input crossbar throughput is $1 - \frac{1}{N}$ of ideal case.

For a large crossbar throughput approaches $1 - \frac{1}{e}$ of ideal case.

Comparison to Bus

Much faster. Performance considered ideal.

Also more expensive, cost considered worst-case.

Direct Networks

Structure

Consist of *nodes* and *links*.

Link: carries data between two nodes.

Node: connects links to each other and to a PE.

Each node attaches to one PE and one or more links.

Related Definitions

If a link attaches to a node it is said to be *incident* to the node.

Two links are said to be *adjacent* if they are incident to the same node.

The *degree* of a node is the number of links incident to the node.

The *degree* of a network is degree of the node with the highest degree.

There are a great variety of direct networks.

Comparison to Bus

A link carries data from one node to exactly one other.
(Unlike the data-carrying medium in a bus, which carries data to all outputs.)

PEs do not directly connect to links.
instead they go through nodes.

Routing

Path between two nodes: a sequence of links connecting the two nodes in which:

The first link is incident to one node,...

...the last link is incident to the other node,...

...and all pairs of consecutive links in the sequence are adjacent.

To *route* [a request for a network] is to to find a path for the request in the network.

Routing in bus and crossbars is trivial.

Some direct networks are easy to route.

Other direct networks are hard to route.

Message Traversal (Flow Control)

Flow control: the method used to move data from node to node.

Two families of flow control methods:

- *Circuit Switching*

  A path is reserved for a request.

  Request has exclusive use of path.

  Analogous to a telephone system...

  ...a phone call is like a request; conversation is like a message.

- *Packet Switching*

  No path reserved.

  Message moves from node to mode.

  Message might be delayed at a node.

  Analogous to a postal system...

  ...a letter is like a message.

  There are several types of packet-switching flow-control methods.

Packet switching is much more common.

Circuit switching primarily used in multistage networks.

Each family requires a different node design.

Generic Node Contents

Nodes may contain the following items:

- *Buffer:* storage area for messages.
- Unit of storage in buffer is called a *slot*.
- Crossbar, used to connect links and PE to buffer.
- *Controller,* controls operation of node.

Message Structure

Messages are divided into *packets.*

By definition, a packet can be transfered by a network in one step.

The exact definition of step depends upon the network.

Packet Switching

Typical Packet-Switching Node

Each link has a multiple-slot buffer.

Buffer holds packets entering the node.

Buffer connects to crossbar, crossbar to links or PE.

Packet movement synchronized with a *clock*.

During a clock cycle:

- A packet starts in one buffer,
- moves through a crossbar,
- to a link,
- through the link to another node,
- and into a buffer.

In some networks, more than one cycle might be needed for this.

Packets will be *blocked* when more than one simultaneously needs to use the same link.

Common Packet-Switching Flow-Control Methods

Store and Forward:

- all packets in a message must arrive at a node before the first packet can leave the node.

  This simplifies design but slows large messages.

Wormhole Routing:

- all packets of a message follow the same path,
- a packet can leave a node when a link is available,
- and the first packet of any message has lower priority than non-first packets when arbitrating for a crossbar output.

  The last condition keeps messages together.

Typical Circuit-Switching Node

Controller helps build and tear down paths through network.

Buffer holds one packet per link.

(Since paths are reserved, packets are never delayed, so only one slot per link is needed.)

Comparison

Circuit switching may be better when messages are large.

Additional variations might be covered later in the semester.

Direct Network Metrics

Performance of a direct network depends upon how far messages have to go.

"How far" is quantified by the following metrics:

Distance between two nodes: the number of links on the shortest path between the two nodes.

Distance between u and v denoted: $d_{u,v}$.

Diameter of a network: the maximum of distances over all pairs of nodes.

Diameter denoted D.

In other words, the maximum number of links that must be traversed by any request.

Average distance: average of distances taken over all distinct node pairs in a network.

Average distance denoted $\overline{d}$.

Bisection width: the smallest number of links crossing a line dividing the nodes of a direct network into two parts, with the number of nodes in each part differing by at most one.

Bisection width denoted BW.

Latency-Related Performance Measures

Latency: The time elapsed from when a message starts entering a network to when the message finishes leaving the network.

Latency denoted L.

Latency is divided into three components:

Propagation time, the time the first packet of a message would take to travel from its source to destination in an empty network.

Waiting time, the time spent **waiting** in buffers.

Pipeline delay, the number of packets in the message minus one.

Communication Patterns

Recall, request (x, y) is a need to connect input x to output y.

A *connection assignment* is a set of requests simultaneously presented to the network.

Different connection assignments favor different networks.

Cost Models

Goal: Estimate cost of network.

Method:

- Decide which parts of the network to include.

  Include links and crosspoints, don't include control logic.

- Count number of each type of part in the network.

- Optionally, assign a cost to each part.

  Link costs 2, crosspoint costs 1.

Parts Frequently Included in Cost

Links.

Crosspoints.

Buffer slots.

Part Rarely Included in Cost

Control logic. (Difficult to estimate cost.)

Multistage Networks

Abbreviation: MIN.

Fourth type of network.

Also called indirect and dynamic (rare) networks.

Parts:

- *Cells*, like nodes except do not necessarily connect to processors.
- *Links*, same as links in direct networks except data only moves in one direction.

Structure:

- Cells are arranged in *stages*.
- Stages[1] numbered 0, 1, 2, . . . , (with consecutive integers starting with zero).
- Cell in stage i connected by link to cell in stage j only if $i < j$.
- *Network inputs* connect to cells in stage 0.
- *Network outputs* connect to cells in highest-numbered stage.
- Messages enter through inputs and exit through outputs.

[1] Without loss of generality.

Comparison to Direct Networks

Distance

With direct networks, some PEs are at distance 1.

With MINs, most or all PEs are same distance, usually $\log_x N$.

Flexibility

Some believe MINs are more flexible since PEs not "part of" topology.

Certain connections might be easier on a MIN.

Conventional Wisdom

For small systems, buses are best.

For medium-size systems (between 30 and 500 PEs) direct networks probably best.

For future large systems: maybe MINs.

For non-computer communication switching: MINs.

Multistage Network Functions

In addition to switching MINs can:

- Sort.

  Request contains a *key* instead of destination.

  At outputs, messages appear in sorted order.

- Reduction, Combining.

  Request includes an opcode.

  Network performs some time of operation on messages.

  For example, taking a sum.

- Synchronization.

  Messages used to alert a PE when other PEs have reached a certain point.

- Counting.

  Network can assign PEs a unique number. (This is more useful than it may sound.)