Palestine Polytechnic University
Collage of Administrative Science and Informatics
Department of Information Technology

Network Simulation

Project team:

Aya Ibrahim Al-Saeid | Mayy Hijazi Abu-Meizer

Supervisor:

Ahlam Qurei'

A final project submitted in partial fulfillment of the requirements for the degree of B.Sc. in Information Technology

2007

Dedication

To our parents for their support and blessing through our lives and especially through the past four months.

And to all our friends that appreciates us and stood beside us who are not cited and whose names may be inadvertently not mentioned…

Aya Al Saied
Mayy Abu-Meizar

Acknowledgment

At this stage we would like to give our thanks and appreciates too many people that stood by us and supported us, to our Dear families.

Also we want to Thank Bssam Al-Hour for his support, Inas Silmy, Sanaa' al Saied ,and Shereen Abu sneneh.

We will never forget our University, IT Department their support. Many thanks to all our friends and teachers for the times and efforts they spend in this project

Aya Al Saied
Mayy Abu-Meizar

Abstract

This project describes a network simulator for a specific topology (tree topology), that measures up the performance of this network using a two ports switches and PCs. The proposed system aim to transmit data packets to specific destinations in the network. The Network tree is created, Database tables are instantiated, and sub-processes are included.

Each Packet is transmitted to its destination; all the information is being stored in the database tables to be calculated after the system clock is completed.

The results being calculated and presented in the tables according to the transmission information.

Table of Content

Dedication	II
Acknowledgment	III
Abstract	IV
Table of content	V
List of tables	VIII
List of figures	IX
Declaration	X
1.0 INTRODUCTION	1
1.1 Overview	1
1.2 Project Hypothesis	1
1.3 Project Objectives	2
1.4 Project Contributions	2
1.5 Project Methodology	2
1.6 Project Domain	2
1.7 Project Pioneer Points	2
1.8 Project Outline	3
2.0 BACKGROUND	
2.1 introduction	5
2.2 What is Computer network	5
2.2.1 Computer network components	5
2.2.2 Computer network protocols	5
2.2.3 Types of Computer network	7
2.3 Local area network (LAN)	7
2.3.1 Router Network	8
2.3.1.1 Network routing	8
2.3.2 Hub Network	9
2.3.2.1 Network Hub types	9
2.3.2.2 Ethernet hub	9
2.3.2.2.1 hub usefulness	10
2.4 Switching Networks	11
2.5 Basic concepts of switching networks	11
2.5.1 Switch strategy	11
2.5.2 The topology	11
2.5.3 Buffer	16

2.6 Proposed Topology	17
2.7 Considerations When Choosing a Topology	18
2.7.1 Why using tree topology	18
2.8 Packet	18
2.8.1 Error detection	19
2.8.2 Host addressing	19
2.9 Network performance	20
2.9.1 Throughput	20
2.9.2 Latency	20
2.9.3 The Traffic Patterns	20
2.10 Simulation Literature Review	21
2.10.1 Network simulation based on switches	21
2.10.2 Network simulation based on routers	21
3.0 The Simulation Program	
3.1 introduction	24
3.2 The Simulation Program	24
3.2.1 Simulator Goal and Objectives	25
3.2.2 The Simulator Assumptions	25
3.2.3 The Simulator Scenario	26
3.3 The Simulation Algorithm	27
3.3.1 General Description	27
3.3.2 Simulator Flow Charts	28
3.3.2.1 Flow chart of simulator	28
3.3.2.2 Flow chart of Address generation	29
3.3.3 The Algorithm Stages	30
3.3.3.1 Stage_1: The Initial Stage	30
3.3.3.1.1 Local table transmission	30
3.3.3.1.2 Buffer Table	31
3.3.3.2 Stage_2: Request Address Generation	31
3.3.3.3 Stage_3: Packet transmission	32
3.3.3.4 Stage_4: Updating Tables	32
3.3.3.5 Stage_5: Final Stage	33
3.4 Running the Program	34

4.0 The Results	
4.1 Introduction	36
4.2 Switch Throughput	36
4.3 Packet Latency	38
4.4 Comparing the proposed network with other networks	39
5.0 Conclusion	
5.1 Conclusion	41
5.2 Achievements	41

List of Tables

Table Number	Table Name	Page Number
3.1	Local and Transmission Table	30
3.2	Buffer Table	31
4.1	Average Result Table	37
4.2	Packet latency	38

List of Figures

Figure Number	Figure Name	Page Number
3.1	Simulator Network Hierarchal	26
3.2	Switch With Two Ports	26
3.3	Flow Chart Of Simulator	28
3.4	Flow Chart Of Request Generating Address	29
4.1	Switch Bandwidth	37
4.2	Switch Throughput	38
4.3	Packet Latency	39
4.4	Packet length	39

Declaration

The research team their by declare that this project has been written by them, and all material not their own work has been identified by real and true references, and they are responsible for anything written inside the project.

Aya Ibraim Al-Saied
Mayy Hijazi Abu-Meizar

CHAPTER ONE
INTRODUCTION

1.1 Overview
1.2 Project Hypothesis
1.3 Project Objectives
1.4 Project Contributions
1.5 Project Methodology
1.6 Project Domain
1.7 Project Pioneer Points
1.8 Project Outline

1.1 Overview

In recent years computer networks are increasingly needed and the demand of high speed switching has risen with the increase in bandwidth available in all types of computer networks [1]. That means the performance of networks should be acceptable by the users.

From this point we aim to provide a simulation program that measures the performance of a virtual network for a local area in order to see how this network will operate when it's being in use ,using switching network that is basically use to forward packets from any input port to any given output port .

1.2 Project Hypothesis

- The study assumes that all hardware devices used in this simulator are Switches and Pc's only.
- The study assumes that the used topology is tree network topology.
- The program will be tested in VC++ environment, where no graphical user interface is presented.
- The study presumes that the database will be updated whenever there are changes in the transmission information.

1.3 Project Objectives

The main goal is to measure the performance of a computer network built on tree topology, based on specific assumptions that the work team agreed on.
The simulation program is designed to achieve the following objectives:

- To test the network performance parameters (throughput and latency) under various packet length.
- To test a switching network performance on a tree topology.

1.4 Project Contributions

The team believes that this project has the following contributions:

1. Testing the performance of the Tree Topology as a computer network.
2. A simulator for the future studies has been designed and built. The simulator design is presented in chapter four.

1.5 Project Methodology

1. Literature survey: an in-depth study of computer networks, their parameters, features and concepts was done before conducting the literature survey. The literature concentrated on the efforts paid to improve the network performance through modifications on the topology levels, and the number of devices used.
2. A detailed study on router and router switch and their mechanism, architecture, and design.
3. A detailed study on network topologies to decide which topology matches the requirements.
4. A simulator was build taking into consideration that it is for tree topology.
5. The simulator was run under various values of network parameters (network size, packet length) to test the throughput and the latency in order to see their effect on the network performance.
6. Results were collected and compared to each other in order to see the alternate performance depending on the user choices.

1.6 Project Domain

This program designed in this project is oriented for small firms, university computer labs, offices and any other places that may use small networks and share an average amount of data.

1.7 Project Pioneer Points

- The project is oriented to one network topology (Tree topology) which combines between two topologies, linear bus and star.
- The project running time is quit average to let a good number of packets to be sent to its destination.

1.8 Project Outline

Following this chapter, **Chapter 2** presents a full detailed background about networks, switches, routers, used network topology and a literature review about switching networks and simulations that been designed for these networks.

Chapter 3 presents and discusses the design of the simulator program; building a prototype of this topology is costly and time consuming. A simulator was designed to simulate the network depending on its topology.

Chapter 4 presents the result of the simulation with figures illustrations for throughput and latency, packet number.

Chapter 5 includes the conclusion of the whole work and the suggested future work.

CHAPTER TWO
BACKGROUND

2.1 Introduction

2.2 What is Computer network?

2.3 Local area network (LAN)

2.4 Switching Networks

2.5 Basic Concepts of Switching Network

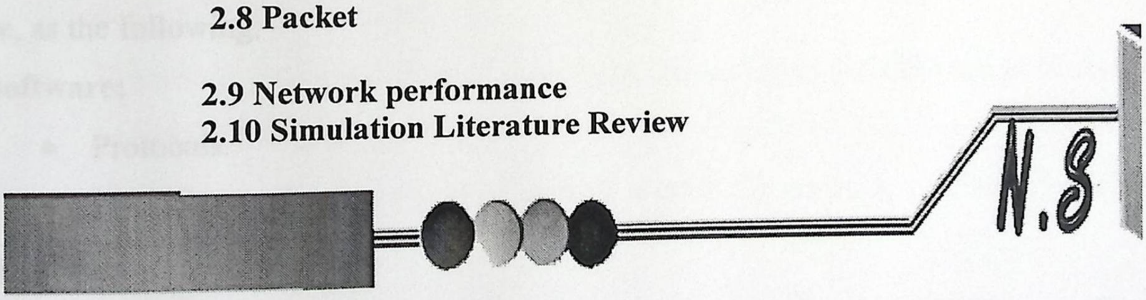
2.6 Proposed Topology

2.7 Considerations When Choosing a Topology

2.8 Packet

2.9 Network performance

2.10 Simulation Literature Review



2.1 Introduction

Computer networks are widely needed and used in offices, schools, hospitals, Universities, many different locations with various areas used so it is needed to have more than one computer network architecture that will be explained in the following sections.

2.2 What is Computer network?

A computer networks (CN) are multiple computers connected together using a telecommunication system for the purpose of communicating and sharing resources, and computer network architecture is the framework for designing and implementing networks.

Generally the computer network must provide cost effectiveness, robustness, and high performance connectivity among a large number of computers, also it must be able to evolve and accommodate changes in both the underlying technologies and the demands imposed by application software.

2.2.1 Computer network components

Generally computer networks consist of two major components software and hardware, as the following:

1. **software:**
   - Protocols.
   - services
2. **hardware:**
   - Transmission technology media and devices.
     1. Point-to-point.
     2. Broadcast.
   - Scale: LAN's, WAN's and MAN's.
   - Topology: star, bus, ring, mesh.
     1. Logical: how hosts communicate across medium.
     2. Physical: the actual layout of medium.

2.2.2 Computer network protocols

Computer networks protocols such as ISO/OSI (International Standard Organization's/ Open System Interconnect), and TCP/IP (Transmission Control Protocol/Internet Protocol) models, are based on a layered architecture. The purpose of each layer tends to offer a communication services to higher level layers. In Figure 3.1 shows the layers for both ISO OSI and TCP\IP which is detailed as the following:

1. **ISO OSI protocol:**
   - Physical: transmission of raw bits onto the communication media.
   - Data link: reliable transmission of frames, flow control, arbitration.
   - Network: packet switching, routing.
   - Transport: prosses-to-prosses channel, node-to-node connection flow control, user services, multiplexing.
   - Session: the protocols necessary to establish and maintain a connection or session between 2 users.
   - Presentation: effective communication of information rather than data, numbers are converted into code, data is compressed.
   - Application: like the e-maile, FTB, and client-server.

2. **TCP/IP protocol:**
   - Application: comprises the specific functions of each application, such as e-mail, FTB, remote login.
   - Transport: provide a flow of data from the source to the destination host, for use of application layer above.
   - Internet: handle movement of packets around the network, routing is handled here.
   - Host to network: handles the network specific details of interfacing to physical communication channel.

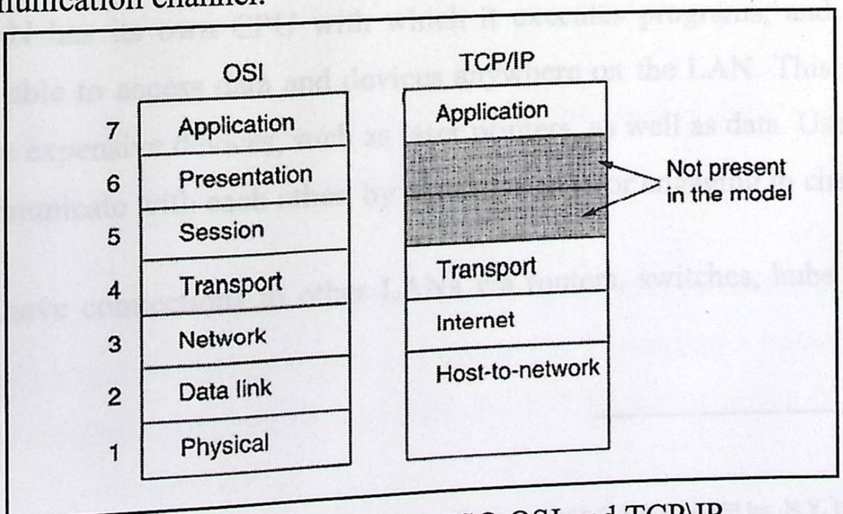


Figure 3.1 Layers for both ISO OSI and TCP\IP

layer tends to offer a communication services to higher level layers. In Figure 3.1 shows the layers for both ISO OSI and TCP\IP Which is detailed as the following:

1. **ISO OSI protocol:**
   - Physical: transmission of raw bits onto the communication media.
   - Data link: reliable transmission of frames, flow control, arbitration.
   - Network: packet switching, routing.
   - Transport: prosses-to-prosses channel, node-to-node connection flow control user services, multiplexing.
   - Session: the protocols necessary to establish and maintain a connection (a session between 2 users.
   - Presentation: effective communication of information rather than data, numbers are converted into code, data is compressed.
   - Application: like the e-maile, FTB, and client-server.

2. TCP/IP protocol:
   - Application: comprises the specific [illegible] mail, FTB, remote login.
   - Transport: provide a flow of data [illegible] use of application layer in the
   - Internet: handle [illegible] here.
   - Host to network: handle [illegible] communication [illegible]

[illegible] a computer network covering a small geographic area, like [illegible] Current LANs are most likely to be based on switched [illegible] technology [1], LANs are capable of transmitting data at very fast rates, [illegible] can be transmitted over a telephone line; but the distances are limited, [illegible] on the number of computers that can be attached to a single LAN. [illegible] late 1970s and used to create high-speed links [illegible] computers at one site. Of many competing systems created at [illegible] and ARCNET were the most popular.

[illegible] LANs connect workstations and personal computers. Each node (individual [illegible]) in a LAN has its own CPU with which it executes programs, and computing [illegible] it also is able to access data and devices anywhere on the LAN. This means that [illegible] users can share expensive devices, such as laser printers, as well as data. Users can also [illegible] LAN to communicate with each other, by sending e-mail or engaging in chat sessions.

LANs may have connections to other LANs via routers, switches, hubs or Internet [illegible]ections.

---

[1] Kai Hwang,"Advanced Computer Architecture: Parallelism, Scalability, Programmability", McGraw-Hill Inc.,N.Y, U.S.A,1993.

2.2.3 Types of Computer network

CN are categorized under two main categories according to the transmission method:

1. Broadcast: all computers are connected by a single continuous communication channel.
2. Switched: data is routed to it's destination by a sequence of point-to-point or node-to-node links.

The performance of a computer network is measured by Bandwidth (number of bits transmitted per unit time), and latency (how ling it takes one bit to propagate from one end network to the other), cost is measured by the number of devices needed to built the network and the cost of each unit.

2.3 Local area network (LAN)

A **local area network** is a computer network covering a small geographic area, like home, office, or group of buildings, Current LANs are most likely to be based on switched IEEE 802.3 Ethernet technology [1], LANs are capable of transmitting data at very fast rates, much faster than data can be transmitted over a telephone line; but the distances are limited, and there is also a limit on the number of computers that can be attached to a single LAN.

The first LANs were created in the late 1970s and used to create high-speed links between several large central computers at one site. Of many competing systems created at this time, Ethernet and ARCNET were the most popular.

Most LANs connect workstations and personal computers. Each node (individual computer) in a LAN has its own CPU with which it executes programs, and computing power but it also is able to access data and devices anywhere on the LAN. This means that many users can share expensive devices, such as laser printers, as well as data. Users can also use the LAN to communicate with each other, by sending e-mail or engaging in chat sessions.

LANs may have connections to other LANs via routers, switches, hubs or Internet connections.

[1] Kai Hwang,"Advanced Computer Architecture: Parallelism, Scalability, Programmability", McGraw-Hill Inc.,N.Y, U.S.A,1993.

The following characteristics differentiate one LAN from another:

- **Topology :** The geometric arrangement of devices on the network. For example, devices can be arranged in a ring or in a straight line.
- **Protocols :** The rules and encoding specifications for sending data. The protocols also determine whether the network uses a peer-to-peer or client/server architecture.
- **Media :** Devices can be connected by twisted-pair wire, coaxial cables, or fiber optic cables. Some networks do without connecting media altogether, communicating instead via radio waves.

2.3.1 Router Network

A **router** is a computer networking device that buffers and forwards data packets across an internet work toward their destinations, through a process known as routing. Generally, the router performs two main functions; Control path routines, and Data path control (switching).

Routers also classify packets and perform control actions on the packets; it performs Layer 3 switching and sometimes maintains statistical data on the data –flow typically, packets are received at inbound network interface; they are then processed by the processing module (CPU), possibly stored in the buffering module. The packets are then forwarded through the switching fabric to outbound interface that transmits the packet to the next hop router.

2.3.1.1 Network routing

The routing algorithm determines which routes the packet follow through the network. In this thesis are presented the following routing algorithm.

1. Wormhole routing or cut-through routing: at the beginning each switch examines the header, decides where to send message, and then starts forwarding it immediately.
2. Store and forwarding: each switch waits for the full packet to arrive in the switch before it sent on to the next switch.

2.3.2 Hub Network

Hub (sometimes referred to as a concentrator) refers to a networking component which acts as a convergence point of a Network allowing the transfer of data packets. Hubs are commonly used to connect segments of a LAN. Hubs work at the physical layer (layer 1) of the OSI model. The device is thus a form of multi port repeater.

A hub works by duplicating the data packets received via one Port and making it available to all ports, therefore allowing data sharing between all devices connected to the hub, so that all segments of the LAN can see all packets.

Hubs do not manage any of the traffic that comes through them, and any packet entering any port is broadcast out on every other port (every port other than the port of entry). Since every packet is being sent out through every other port, packet collisions result--which greatly impedes the smooth flow of traffic.

2.3.2.1 Network Hub types

1. **Passive hub**: simply allows the data packets to flow through it, as a conduit for the data, enabling it to go from one device (or segment) to another.

2. **Manageable hub:** allows the data transfer to be monitored and the ports to be configured individually, it includes an additional feature that enables an administrator to monitor the traffic passing through the hub and to configure each port in the hub. Manageable hubs are also called intelligent hubs.

3. **Switching hub:** reads the destination address of each packet and then forwards the packet to the correct port.

2.3.2.2 Ethernet hub

An Ethernet hub or concentrator is a device for connecting multiple twisted pair or fiber optic Ethernet devices together, making them act as a single segment .Ethernet hubs are also responsible for forwarding a jam signal to all ports if it detects a collision.

A hubed Ethernet network behaves like a shared-medium, that is only one device can successfully transmit at a time and each host remains responsible for collision detection and retransmission. An Ethernet hub, or repeater, is a fairly unsophisticated broadcast device.

Most hubs detect typical problems, such as excessive collisions on individual ports, and partition the port, disconnecting it from the shared medium. Thus, hub-based Ethernet is generally more robust than coaxial cable-based Ethernet, where a misbehaving device can disable the entire segment. Even if not partitioned automatically, a hub makes troubleshooting easier because status lights can indicate the possible problem source or, as a last resort, devices can be disconnected from a hub one at a time much more easily than a coaxial cable. They also remove the need to troubleshoot faults on a huge cable with multiple taps.

2.3.2.2.1 Hub usefulness

Historically, the main reason for purchasing hubs rather than switches was price. This has largely been eliminated by reductions in the price of switches, but hubs can still be useful in special circumstances:

- Some computer clusters require each member computer to receive all of the traffic going to the cluster. A hub will do this naturally; using a switch requires implementing special tricks.
- When a switch is accessible for end users to make connections, for example, in a conference room, an inexperienced or careless user (or saboteur) can bring down the network by connecting two ports together, causing a loop. This can be prevented by using a hub, where a loop will break other users on the hub, but not the rest of the network. (It can also be prevented by buying switches that can detect and deal with loops, for example by implementing the Spanning Tree Protocol.)

2.4 Switching Networks

In this section, the main concepts of switching networks are introduced, and a literature survey of research in this field is conducted. The literature survey concentrated on the development in switching network topologies and performance. The term "Topology" means the physical interconnection structure of the network elements, namely, the terminal nodes, the switches and links. Performance basically means the amount of data the network can transport in time unit. In the following section the basic concepts of topology and performance are explained.

2.5 Basic Concepts of Switching Network

2.5.1 Switch Strategy

Switching strategy determines how the data in a message traverses its rout through the network. There are basically two switching strategies, circuit switching and packet switching. In circuit switching the path from source to destination must be established and reserved first, before the message is transferred over the circuit. Subsequently the connection is then removed. The alternative is packet switching in which the end nodes send messages by breaking them up into a sequence of packets, which are individually routed from the source to the destination. Each packet contains route information examined by switching elements to forward it correctly to its destination; packet switching typically allows better utilization of network resources. A combination of the two strategies will be used in this simulator. The packet will be used as a data transfer unit to be sent through the network.

2.5.2 The Topology

The network topology decides the physical structure of the way the nodes, the links and the switches are organized and connected. The basic building block of these networks is the crossbar switch. The crossbar is a non-blocking switch. Non-blocking means that any permutation of inputs and outputs can be connected without interfering with each other. However, not more than one input port can be connected to the same output at the same time. Usually the number of output ports is equal to the number of input ports. Figure 2-1 shows a schematic of the crossbar connection.

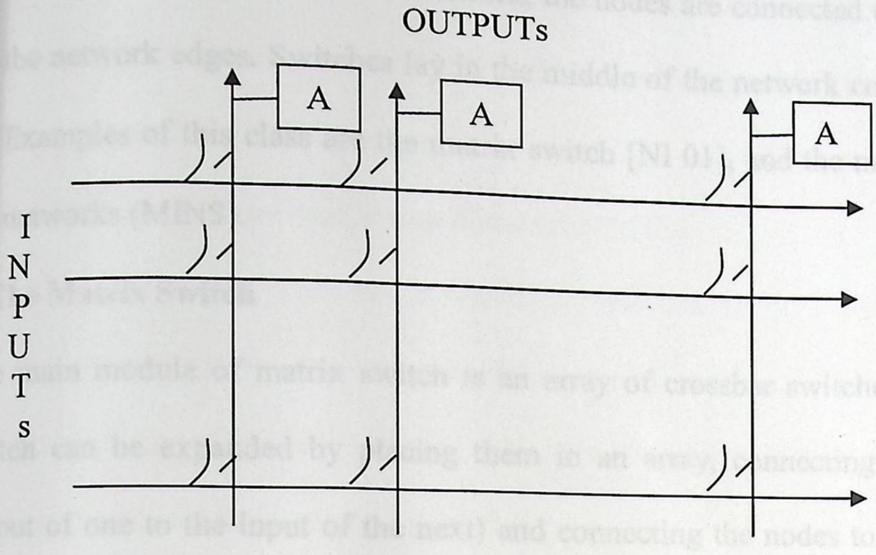


Figure 2-1: The crossbar switch

In figure 2-1 any input can be connected to any output by closing the switch on the cross-point between the input and the output. If more than one input is trying to connect itself to the same output, the arbiter (A) of the output selects one input and rejects the others. So not more than one input can be connected to the same output.

Expanding the crossbar switch to accommodate more inputs and outputs proved to be costly and complicated [HWA 93, Wilk 96]. Doubling the number of inputs and outputs implies that the cost grows four times. In other words the cost and complexity of the crossbar grows as a function of N^2, where N is the number of inputs which is usually equals the number of outputs. So other less cost topologies are needed.

2.5.2.1 Expansion Topologies

The expansion topologies of the crossbar switch falls into two main classes [Haas 98]:

1. The Direct Networks: In these networks the nodes are directly connected to the switches of the network. The node means the computer and its interface to the network. Each switch and its nodes represent a module in the network. Modules are arranged in a matrix fashion and enough links are available between each module and its four neighbors. Example is the 2D-Grid and Torus networks.

2. The Indirect Networks: In this network, the nodes are connected to the switches laying on the network edges. Switches lay in the middle of the network connect the edge switches. Examples of this class are the matrix switch [NI 01], and the multistage interconnection networks (MINS).

a. The Matrix Switch

The main module of matrix switch is an array of crossbar switches of the 4X4 size. This switch can be expanded by placing them in an array, connecting the internal edges, (the output of one to the input of the next) and connecting the nodes to two of the four external edges of the array [NI 01]. This expansion method is as costly as the crossbar switch expansion especially for networks with large number of nodes.

b. The 2D-Grid Network

In this network the nodes are directly connected to their switch modules. The modules are arranged in a matrix fashion. Enough links are provided between each module and its four neighbors. However, not all links of the switch module are used for connecting the nodes. Part of the switch module links are used to provide connections with the neighboring modules. Figure 2-2 shows how the 2D-Grid is implemented in Macramé project [Haas 98].

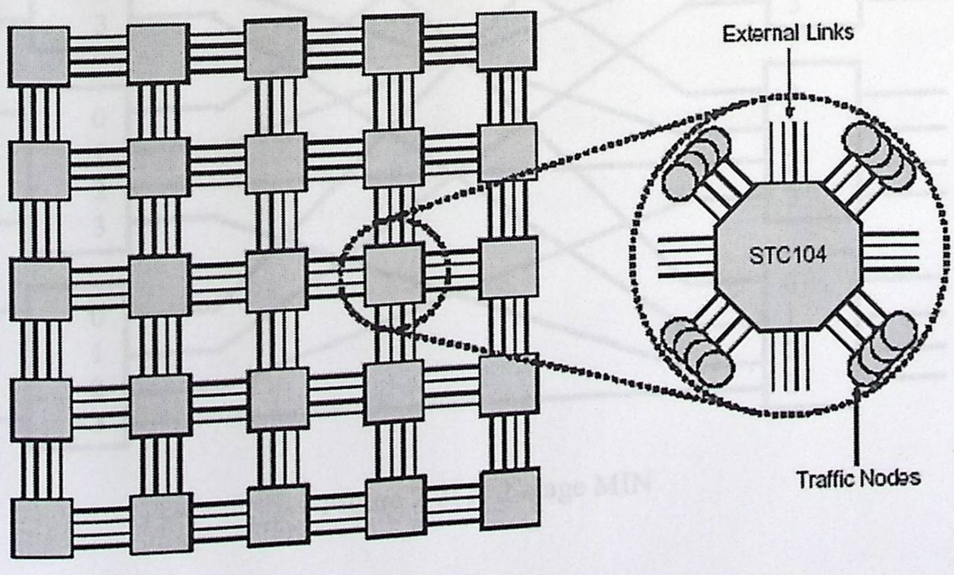


Figure 2-2: The 2D-Grid expansion topology

From the above figure, it is clear that half the 32 links of the STC104 module are used to accommodate 16 nodes, and the other half is used to provide 16 connection with the four neighboring modules, four links for each.

c. The Multistage Interconnection Networks (MIN)

In 1953, Charles Clos published a technical paper describing a method to reduce the cross-points count through the multistage switching technique [Yang 99, HWA 93, Wilk 96]. Since that time the age of the MIN started. Different connection fabrics, and different switch sizes were proposed and implemented since then.

In this topology the switching modules are arranged in columns. The outputs of each column are connected to the inputs of the next in a shuffle fashion to provide a non-blocking connection between the inputs of the first column and the outputs of the last column. Figure 2-3 shows a 2-stage MIN built of 4X4 switch module, which connects 16 inputs to 16 outputs.

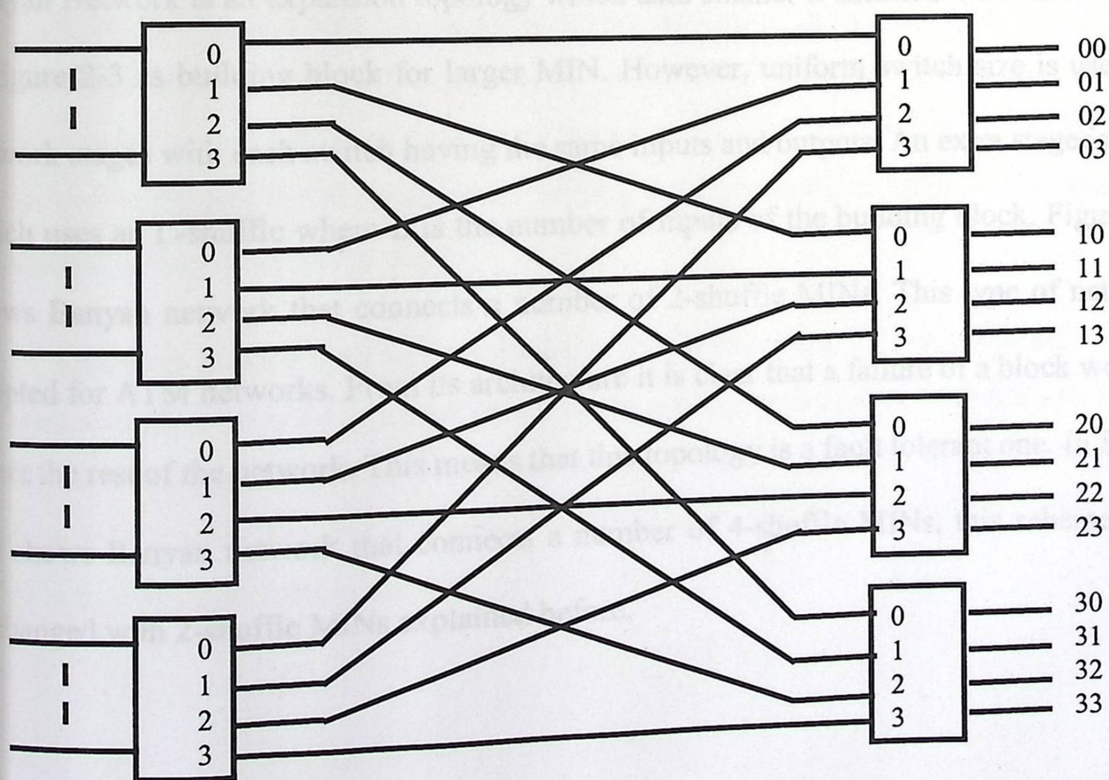


Figure 2-3: A 2-stage MIN

Varieties of MINs can be found in the literature. The most two used MIN network topologies are shown in the following:

Clos Network is a MIN that allows the usage of various sizes of switches, with general nxm switch. This topology provides the network designers with flexibility. Figure 2-4 show a Clos network with nxm switch size and r stages.

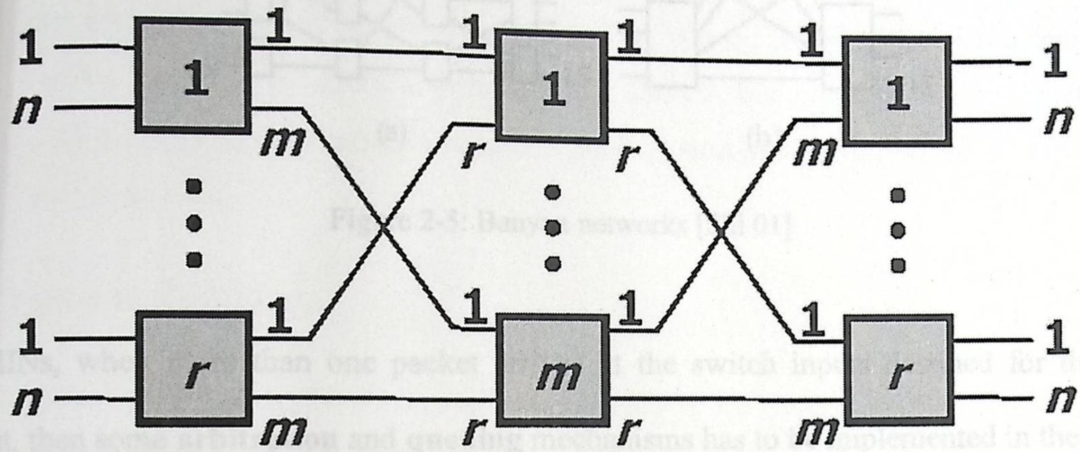


Figure 2-4: Clos network u(n,m,r) network

Banyan Network is an expansion topology which uses smaller n-shuffled MIN networks like in figure 2-3 as building block for larger MIN. However, uniform switch size is used in all network stages with each switch having the same inputs and outputs. An extra stage is needed which uses an L-shuffle where L is the number of inputs of the building block. Figure 2-5-a shows Banyan network that connects a number of 2-shuffle MINs. This type of network is adopted for ATM networks. From its architecture it is clear that a failure of a block would not affect the rest of the network. This means that this topology is a fault tolerant one. In figure 2-5-b shows Banyan network that connects a number of 4-shuffle MINs, this scheme can be exchanged with 2-shuffle MINs explained before.

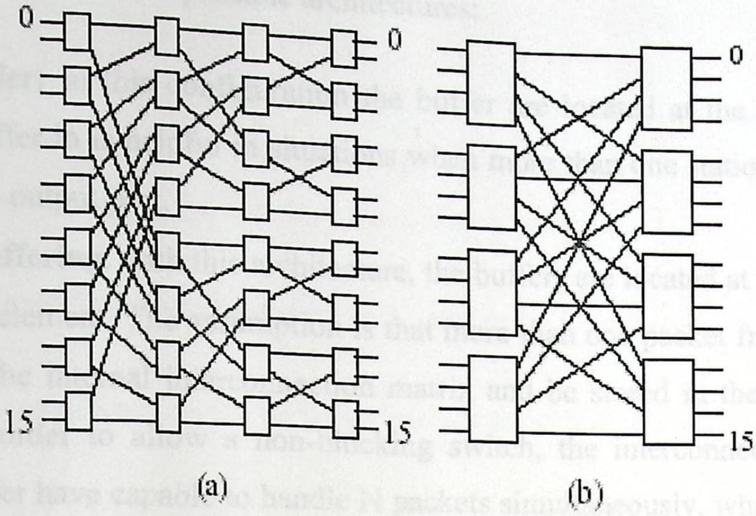


Figure 2-5: Banyan networks [Shi 01]

In MINs, when more than one packet arrives at the switch inputs destined for the same output, then some **arbitration** and **queuing** mechanisms has to be implemented in the switch.

2.5.3 Buffer

A buffer is a data area shared by hardware devices or program processes that operate at different speeds or with different sets of priorities. The buffer allows each device or process to operate without being held up by the other. In order for a buffer to be effective, the size of the buffer and the algorithms for moving data into and out of the buffer need to be considered by the buffer designer.

This term is used both in programming and in hardware. In programming, buffering sometimes implies the need to screen data from its final intended place so that it can be edited or otherwise processed before being moved to a regular file or database. Buffers are commonly used when burning data onto a compact disc, where the data is transferred to the buffer before being written to the disc.

For buffer there are two possible architectures:

1. **Input Buffer:** in this configuration the buffer are located at the input ports of the switch. Buffer in is helpful in situations when more than one station is trying to send to the same output port.
2. **Output Buffering**: with this architecture, the buffers are located at the output ports of the switch element. The assumption is that more than one packet from the input ports can cross the internal interconnection matrix and be stored in the same output port buffer. In order to allow a non-blocking switch, the interconnection network and output buffer have capable to handle N packets simultaneously, where N is number of switches ports.

2.6 Proposed Topology

A tree topology combines characteristics of linear bus and star topologies. It consists of groups of star-configured workstations connected to a linear bus backbone cable, as shown in figure 3.5.

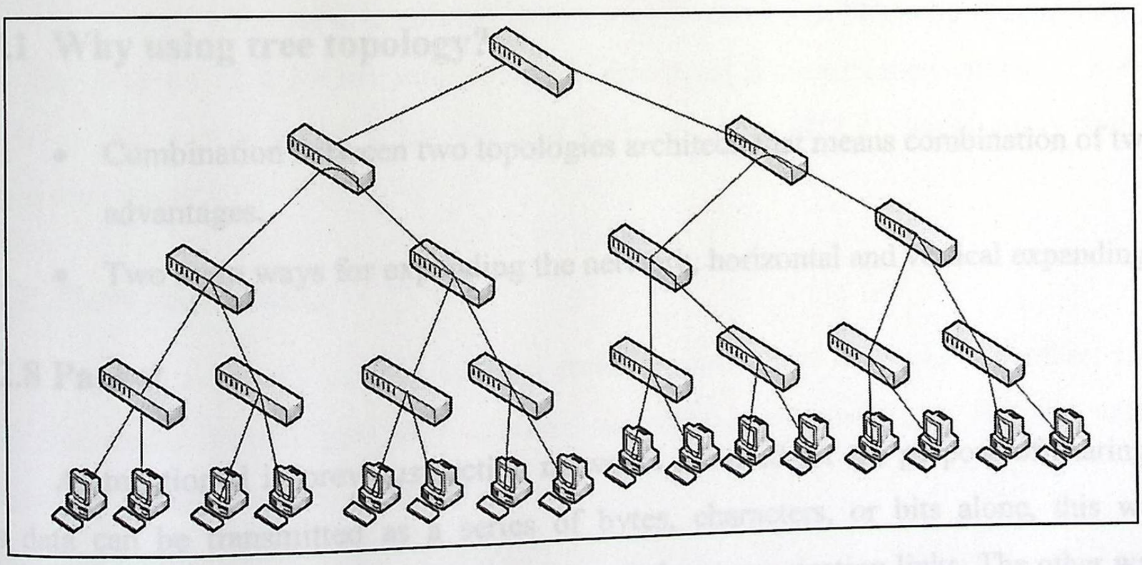


Figure 3.6 Tree Topology

Advantages

- Point-to-point wiring for individual segments.
- Supported by several hardware and software venders.
- Allow for the expansion of an existing network.
- Enable schools/labs to configure a network to meet their needs.

Disadvantages

- Overall length of each segment is limited by the type of cabling used.
- If the backbone line breaks, the entire segment goes down.
- More difficult to configure and wire than other topologies.

2.7 Considerations When Choosing a Topology:

There are many considerations to be taken into account when choosing a topology:

- **Money**. Where the linear bus network may be the least expensive way to install a network, because you do not have to purchase concentrators.
- **Length of cable needed**. The linear bus network uses shorter lengths of cable.
- **Future growth**. With a star topology, expanding a network is easily done by just adding another concentrator.
- **Cable type**. The most common cable in schools is unshielded twisted pair, which is most often used with star topologies.

2.7.1 Why using tree topology?

- Combination between two topologies architect, that means combination of two advantages.
- Two main ways for expanding the network, horizontal and vertical expanding.

2.8 Packet

As mentioned in previous section networks are used for the purpose of sharing data this data can be transmitted as a series of bytes, characters, or bits alone, this way of transition is used in the traditional point-to-point telecommunication links. The other way for transmitting data is by using a packet, which is a formatted block of information carried by a computer network. When data is formatted into a packet, the network can transmit longer messages more efficiently and reliably.

A packet consists of three elements: the first element is a header, which marks the beginning of the packet; the second element is the payload, which contains the information to be carried in the packet; the third element is a trailer, which marks the end of the packet.

So if you considered the packet to be like a letter: the header is like the envelope, and the data area is whatever the person puts inside the envelope. The difference is that some networks can break a larger packet into smaller packets when necessary, but these smaller data elements are still formatted as packets.

Different communications protocols use different conventions for distinguishing between the header, data, and trailer elements and for formatting the data. In Binary Synchronous Transmission, the packet is formatted in 8-bit bytes, and special characters are used to delimit the different elements. Other protocols, like Ethernet, establish the start of the header and data elements by their location relative to the start of the packet. Some protocols format the information at a bit level instead of a byte level [ ].

A network design can achieve two major results by using packets: error detection and multiple hosts addressing.

2.8.1 Error detection

It is more efficient and reliable to calculate a checksum or cyclic redundancy check over the contents of a packet than to check errors using character-by-character parity bit checking. The packet trailer often contains error checking data to detect errors that occur during transmission.

2.8.2 Host addressing

Modern networks usually connect three or more host computers together; in such cases the packet header generally contains addressing information so that the packet is received by the correct host computer.

In complex networks constructed of multiple routing and switching nodes, like the ARPANET and the modern Internet, a series of packets sent from one host computer to another may follow different routes to reach the same destination. This technology is called packet switching.

2.9 Network performance

In this section the measures of network performance are defined. The performance of a switching network can be characterized by throughput and latency. These definitions and concepts were agreed with those used by many simulation projects.

2.9.1 Throughput

The per-node throughput is the number of data bytes received or transmitted by an end-node in a given period of time. The total network throughput is the aggregate transmit or receive rate of all nodes.

2.9.2 Latency

The network latency is defined as how long it takes one bit to propagate from one end network to the other.

The latency is the sum of two components:

- Switching latency;
- Transmission latency;

The switching latency is proportional to the number of switches a packet has to traverse. The transmission latency is the time to transmit a full packet, is proportional to the packet length.

2.9.3 The Traffic Patterns

The performance of a switching network will also be a function of the traffic pattern used. The following traffic parameters influence the network latency and throughput:

- Packet length;
- Packet destination;

Each node sends fixed length packets to random destinations. The destinations are chosen randomly.

2.10 Simulation Literature Review

This field, network simulation has become an active research area. There are a lot of researches made in this domain that can be classified into a few general categories. In this chapter we will review the previous works that were made for network simulation.

Researches done in this domain could be categorized into two categories, one category can be termed as Network simulation based on switches; the other is Network simulation based on routers.

2.10.1 Network simulation based on switches

There are a lot of researches that have been dedicated in this domain like SWARC by Katevenis [1]. SWARC addresses the architecture and design of the switching and routing infrastructure for all kinds of networks and digital systems including electronic & optical switching and their seamless integration, routers and switches for high-performance networks, and switches for WAN, MAN, LAN, storage area, system area, embedded system networks, (multi-) processor-memory interconnects, and networks-on-a-chip. SWARC aims to show how to cost-effectively build this vital switching infrastructure for future communication networks, at all of the above scales, striving to unify the concepts and the architectures, and thus to allow the reuse of design in this wide and growing spectrum of technologies and applications. Minkenberg et al [2] derive new design factors for switch fabrics. For instance, they argue that the packet round-trip transmission time *within* the fabric has become a major design parameter. Furthermore, they observe that high-speed fabrics have become extremely dependent on serial I/O technology that is both high speed *and* high density. Finally, they conclude that in developing the architecture, packaging constraints must be put first and not as an afterthought, which also applies to solving the tremendous power consumption challenges.

2.10.2 Network simulation based on routers

Many researches dedicated in network simulation using routers. Liu and Andrew [3] presents Large-scale network simulation, this extensive large-scale experiments with profile-based load balance (PROF) on flat-routed (OSPF) networks show that PROF outperforms several other techniques based on topology and static application information.

However, these results and those for multi-AS networks motivate our invention of a new hierarchical technique (HPROF) which clusters network nodes to achieve desired minimum link latency (MLL), a key determinant of simulation parallelism, then applies the graph partitioner. HPROF explicitly controls the tradeoff between simulation efficiency and available parallelism, producing robust and superior performance for large-scale networks, including both single- AS and multi-AS networks. HPROF can improve load imbalance by 40%, and reduce the simulation time by about 50% in 20,000 router simulations executed on 128-node clusters. Similar to this approach is the system that was developed by Hollman, Marti [4], who present a new architecture layout for a real time power system simulator based on distributed cluster of IBM PC-compatible desktop computers. A real time network simulator based on a PC-cluster can successfully cope with the size requirements of growing power systems and the computational demands of fast transient studies. A powerful product has been developed using of the shelf Pentium II 40 MHZ workstations with a commercial real time operating system, standard I\O interfaces, and multi machine networking scheme. Models based on the standard tool for power systems transient simulations. The EMTP program, optimized for real time performance assures accurate simulation results.

CHAPTER THREE
The Simulation Program

3.1 Introduction

3.2 The Simulation Program

3.3 The Simulation Algorithm

3.4 Running the Program

3.1 Introduction

Simulation software considered as analysis tools of systems. Simulation software, or simulator, or simulation system, is computer software developed for simulation of systems. Each simulator is designed to meet a specific system requirements and needs. Simulation is a method for solving a problem in systems, which investigates instead of the real system a model of the system or a virtual created system.

Our Model of simulation is being illustrated and described in details in the next sections.

3.2 The Simulation Program

Many simulation programs have been developed in various programming languages and algorithms in order to meet the needs of Network simulation which we figure out of our own simulation program:

1. Measure the performance of specific network topology.
2. Reduce cost and efforts when building the desired network after using the simulator.

The simulator is programmed in VC++(Visual c) for several reasons, and these reasons can be summarized in the following points:

1. Object oriented programming language, which means the ability of using objects and classes to program.
2. Graphical user interface (GUI) is available to the programmer to give the user the ability to interact with the simulator.
3. Inheritance is the ability to define a new class that inherits the behaviors and code of an existing class.
4. VC can be attached with other soft wares like Visual basic (VB), and Quick Time (QT).

3.2.1 Simulator Goal and Objectives

The main goal is to measure the performance of a computer network built on tree topology, based on specific assumptions that will be presented in section 4.2.2.

The simulation program is designed to achieve the following objectives:

- To test the network performance parameters (throughput and latency) under various packet length and network size that will be achieved in the future work.
- To test a switching network performance on a tree topology.
- To compare the network performance under different packet length.

3.2.2 The Simulator Assumptions

The simulation program described in this project assumes the following points:

1. Packet length is dynamic.
2. A full tree topology associated with 15 switches, and 16 nodes (PC's).
3. Network size 5 levels at maximum.
4. Number of devices in the network is fixed according to the tree branches.
5. Destination address generation is random according to the system.
6. Required database for performance measurement:
    - Packet number.
    - Transmission time.
    - Source device.
    - Destination device.
    - Latency (delay).

3.2.3 The Simulator Scenario

The scenario here means the actual sequence of events that occurs when a node needs the service of the network to transfer a packet to a destination node. The source node can send its own generated packet to a destination node that be chosen randomly or receive packets from other source nodes. Figure 4.1 shows the network topology hierarchal contains switches and nodes. When node x (any randomly chosen node), for example needs to send a packet to any destination node chosen also randomly, the node x checks the buffer of the switch that is directly connected with it, and translate the packet bit by bit. The switch then checks the destination address to determine where to passes the packet to through the ports of the switch.

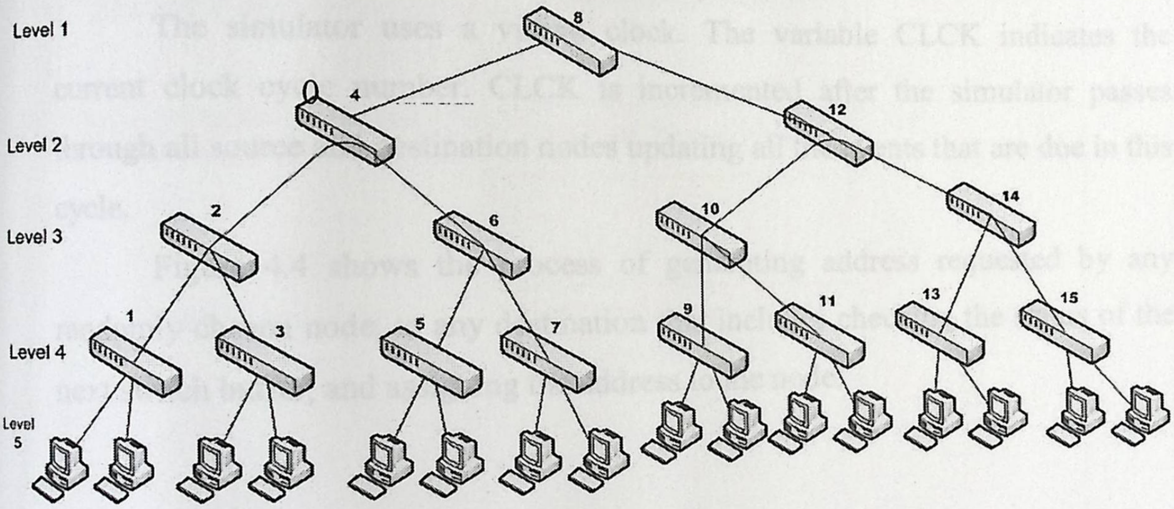


Figure 3.1 – Simulator network hierarchal

Figure 4.2 shows a closer view of the switch illustrating the two ports and the buffers (In and Out) associated to each port.

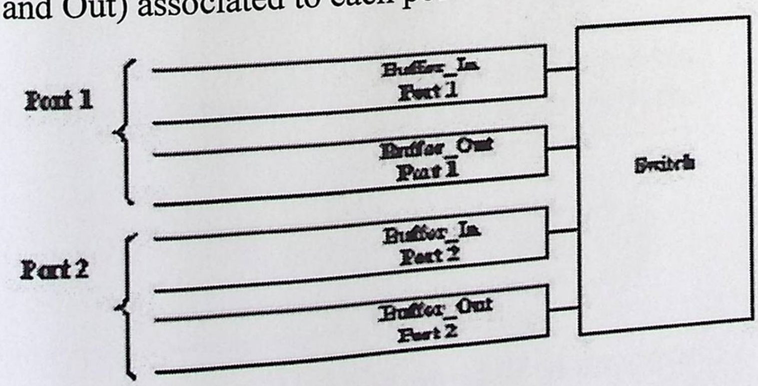


Figure 3.2 – Switch with two ports

3.3 The Simulation Algorithm

3.3.1 General Description

This algorithm represents the behavior of tree network built of switches and nodes. The algorithm takes in consideration that all pasts of the network (nodes, links, and buffers) are working in parallel. Figure 4.3 shows the general algorithm of the simulator.

The simulator is run for a large number of clock cycles ($5 * 10^4$) called the run time (CLCK_MAX). In each clock cycle, all activities which are due in stages 2 to 5 of the algorithm are performed in all parts of the network.

The simulator uses a virtual clock. The variable CLCK indicates the current clock cycle number. CLCK is incremented after the simulator passes through all source and destination nodes updating all the events that are due in this cycle.

Figure 4.4 shows the process of generating address requested by any randomly chosen node, to any destination that includes checking the status of the next switch buffer, and assigning the address to the node.

3.3.2 Simulator Flow Charts[1]

3.3.2.1 Flow chart of simulator

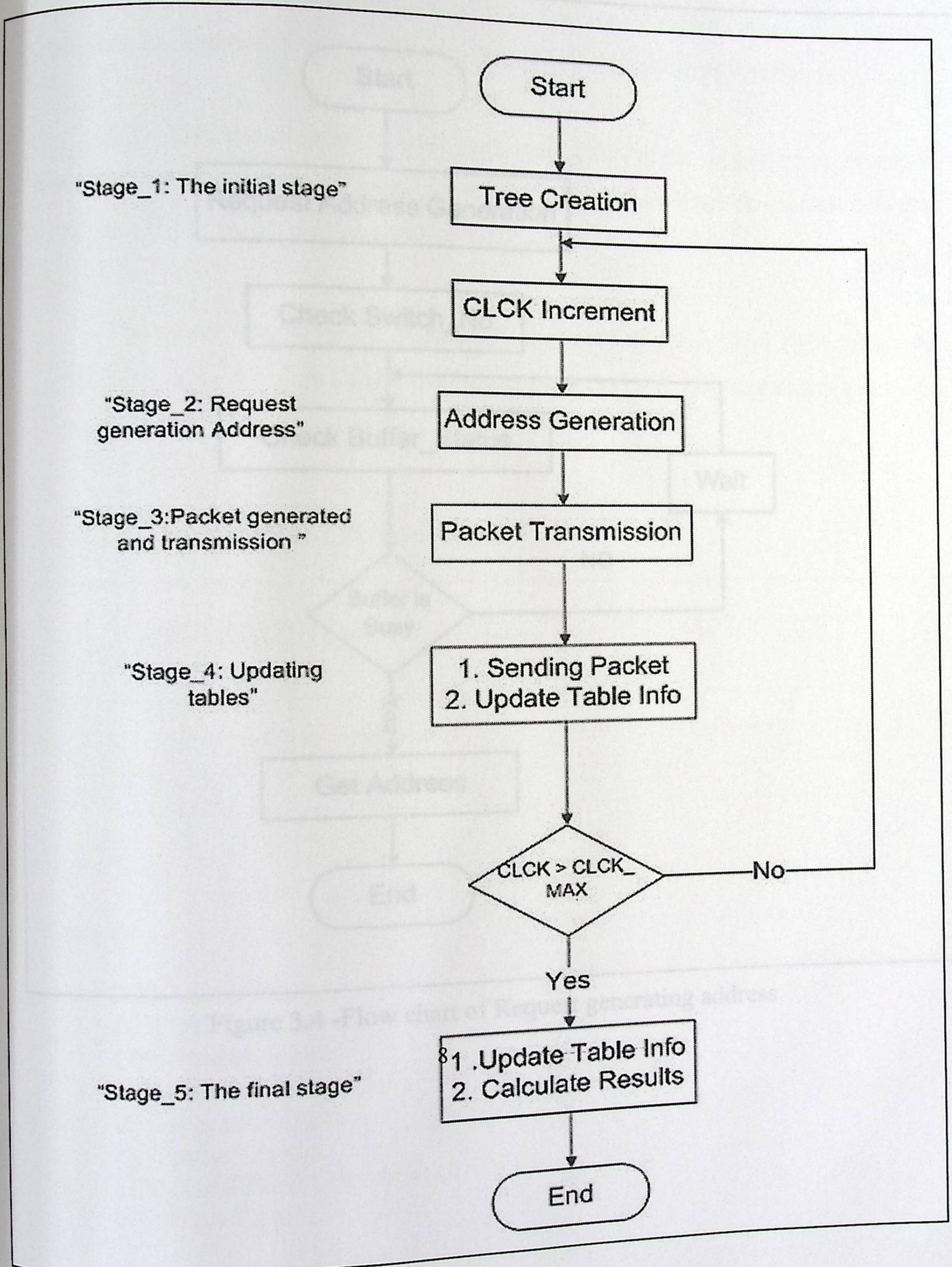


Figure 3.3 -Flow chart of simulator

[1] The diagrams based on Gane and Sarson Symbols

3.3.2.2 Flow chart of Address generation

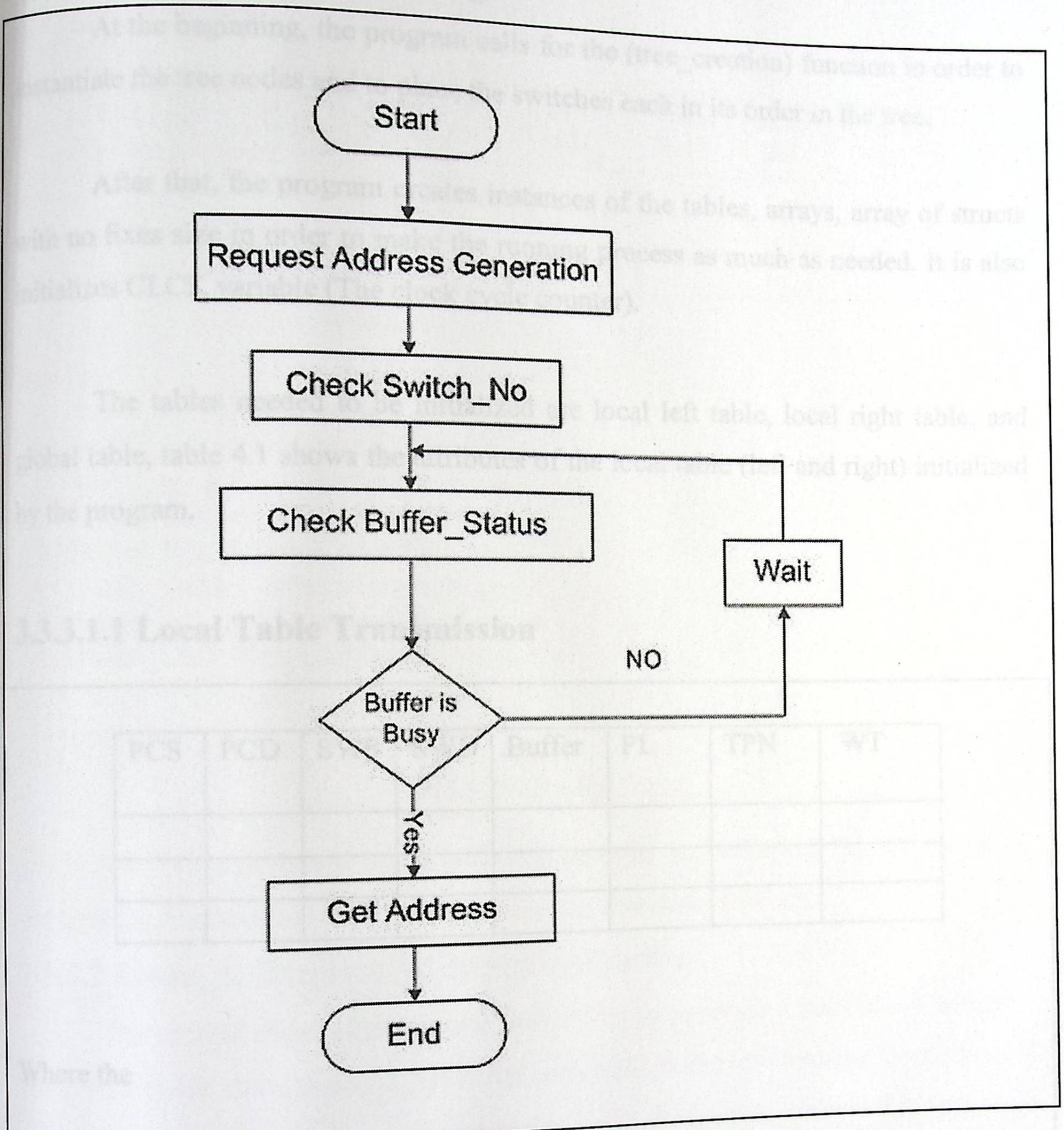


Figure 3.4 -Flow chart of Request generating address

3.3.3 The Algorithm Stages

3.3.3.1 Stage_1: The Initial Stage

At the beginning, the program calls for the (tree_creation) function in order to instantiate the tree nodes and to place the switches each in its order in the tree.

After that, the program creates instances of the tables, arrays, array of structs with no fixes size in order to make the running process as much as needed. It is also initializes CLCK variable (The clock cycle counter).

The tables needed to be initialized are local left table, local right table, and global table, table 4.1 shows the attributes of the local table (left and right) initialized by the program.

3.3.3.1.1 Local Table Transmission

PCS	PCD	SWS	SWD	Buffer	PL	TPN	WT

Where the

1. PCS: PC Source.
2. PCD: PC Destination.
3. SWS: Switch Source.
4. SWD: Switch Destination.
5. Buffer: Switch Buffer.
6. PL: Packet Length.
7. TPN: Total Packet Number.
8. WT: Wait Time.

Table 3.1 – Local and global Transmission Table

3.3.3.1.2 Buffer Table

Buffer_ In	Buffer_ Out	Statu_ In	Status_ Out	Buffe_ In2	Buffe_ Out2	Statu_ In2	Status_ Out2

Where the:

1. Buffer_In: buffer in on left port.
2. Buffer_Out: buffer out on left port.
3. Status_In : left port buffer_in status
4. Status_Out: left port buffer_out status.
5. Buffer_In2: buffer in on right port.
6. Buffer_Out2: buffer out on right port.
7. Status_In2 : right port buffer_in status
8. Status_Out2: right port buffer_out status.

Table 3.2 – Buffer Table

The buffer table shown in table 4.2 explains the buffer_in, buffer_out status, and content, the content is the packet sent to the destination switch.

3.3.3.2 Stage_2: Request Address Generation

The second stage after the tree creation is to generate a destination address , the system clock starts to count ,after that each node in the tree requests the service of addressing ,the address given to these nodes is the destination address that this node will send the packet to.

The address generation process has sub-processes that shown in figure 3.4 .this sub-processes starts with checking the number of the switch that is directly attached with the node that requested the address this checking process have two main aims:

1. To figure out what is the location of the source node (left or right side of the tree).
2. To figure out which table will be called to be the transmission database.

After that check the Buffer_In status of the switch that are directly attached with the source node, if the buffer is free then continue the remain processes of address generation by calling the (getrandom_s) function to generate a switch destination, and then call (getrandom_pc) function to generate a pc destination, if the buffer is busy then count wait time for the node buffer until it get free.

After finishing the checking phase and making sure that the node can send a packet a destination address is give to the source node. This process continues to each node in the tree that request address generation.

3.3.3.3 Stage_3: Packet transmission

When (Buffer_In) flagged as free buffer, packet is being sent bit by bit each bit takes one clock to be transferred from one location to another, when the bits are completely transferred to the switch buffer the program, then the program checks whether this switch is the destination switch, if yes then check the left and right PC's to get which of them is the destination PC ,then check the buffer out that is attached with the destination pc ,if the Buffer_out is busy wait until its get free then start to transmit the packet to it, if this switch is not the destination switch then checks its previous switch and start to transmit the packet if the Buffer_In status is free otherwise wait.

After that checks the previous switch left and right nodes to know which one is the destination switch if the destination switch found then checks its pc's, this checking processes to the switches continues till the destination switch is found.

3.3.3.4 Stage_4: Updating Tables

In this stage, after finishing stage_3 with all its processes of checking Buffer_In , Buffer_out, destination addresses(switch and pc) the information of the transmission phase is being stored and updated in their locations in the tables, these information will be needed when calling the (Calc) function to calculate the results in the final stage.

3.3.3.5 Stage_5: Final Stage (Calculate results)

In this stage the variable (CLCK) is being checked comparing it with the defined (CLCK_MAX), to make a decision; if (CLCK< CLCK_MAX) increment the system clock and continue the process of generating new destination addresses and transmitting more packets. If (CLCK>CLCK_MAX) that means that the clock system is completed, then call for the (Calc) function to calculate results and update tables with these results.

In order to calculate the throughput of the network in (bit/sec), the program takes the sum of totally transferred local packets value from the database table, then takes the sum of totally transferred global packets. The two sums are multiplied by the packet length in bits then divided by the(run time multiplied by the clock period). Equation 3.1 shows the throughput of the system.

$$\text{Bandwidth} = \frac{(\sum TPNL + \sum TPNG) * PL}{(CLCK\_MAX * 10^{-8})} \qquad \text{(bit/sec)}$$

$$\text{Throughput} = \frac{\text{Bandwidth}}{8} \qquad \text{(byte/sec)}$$

$$\text{Delay} = \frac{(\sum TLW + \sum TGW) * 10^{-8}}{(\sum TPNL + \sum TPNG)} \qquad \text{(second)}$$

3.4 Running the Program

For each of entry values, we had to run the program five times in order to obtain the average of the throughput and the latency of that set. Before the five times running, various values were tried in order to reach to the steady state calculations. The run time value (CLCK_MAX = 10^5) was adopted for all tests. we tried to make the program to run the five times for different sets of entry values in order to create tables of throughput and latency versus different values of packet lengths, but unfortunately , the PC stated giving "out of memory" system error message. So we had to run the program as mentioned above. The average values obtained from the five runs, are stored in Excel tables for throughput and latency for various sets of entry values. The result presented in chapter 4 obtained from these tables.

CHAPTER FOUR
RESULTS

4.1 Introductions

4.2 Switch Throughput

4.3 Packet Latency

4.4 Comparing the Proposed Network with Other Networks

4.1 introductions

This chapter presents the results obtained from the simulator described in chapter 3. The performance of proposed network and was compared with other topologies like multi stage Clos n, and 2d grid networks. The result presented here for fixed network size , different packet length.

In order to obtain results run time values (CLCK_MAX = $5*10^5$ / $7*10^5$ / $10*10^5$) discussed in section was entered in each running in the program. In this project all results are measured under the following constrains:

1. 15 switch.
2. 16 computer devices.
3. Random traffic for addresses.
4. All results taken at saturation case, where the saturation mean the maximum achieved throughput.

We start running the simulator by entering a set of values of the run time, the packet length, then running the program to obtain throughput and latency for the entered set. The simulator is run five times for each set for entry values, and then the average of the five results is taken and stored in the relevant excel table.

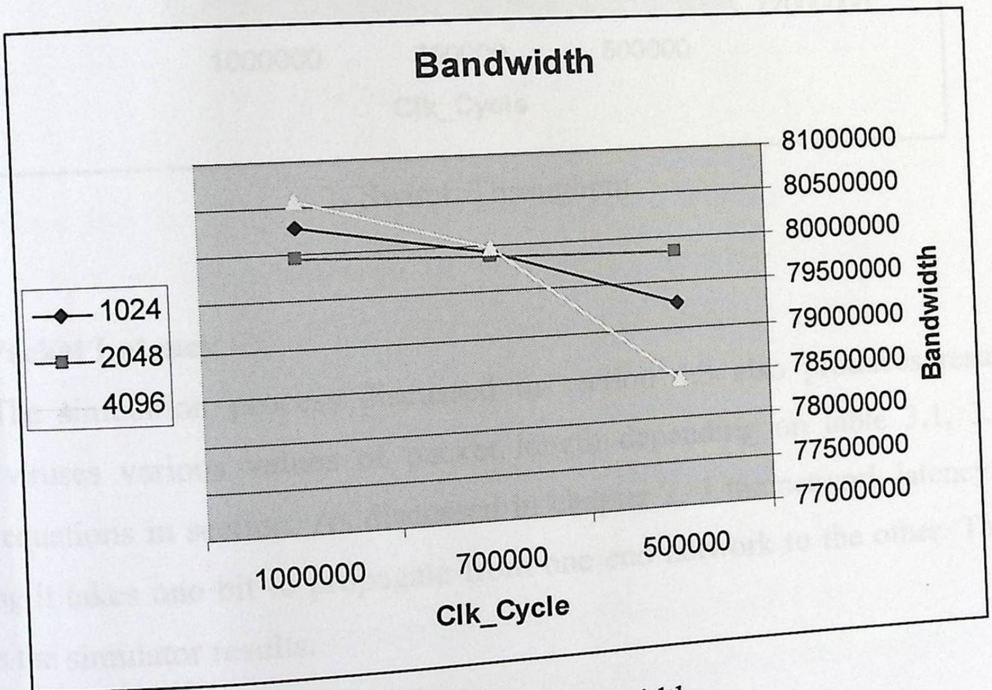
4.2 Switch Throughput

The simulator calculates the throughput of the proposed network from table 3.1, and 3.2 then using equations described in section 3.3.3.5 .the simulator is run for several values of packet length(128,256 and 512 bytes) . The saturation throughput is taken and plotted against the packet length in order to see the effect of the packet length on the throughput see table 4.1 This test also produces results for latency presented in section 4.3.

PL \ Clk_Cycle	$5 * 10^5$	$7 * 10^5$	$10 * 10^5$
1024	79261257	79933440	80289280
2048	79851520	79886629	79969280
4096	78409143	79974400	80568320

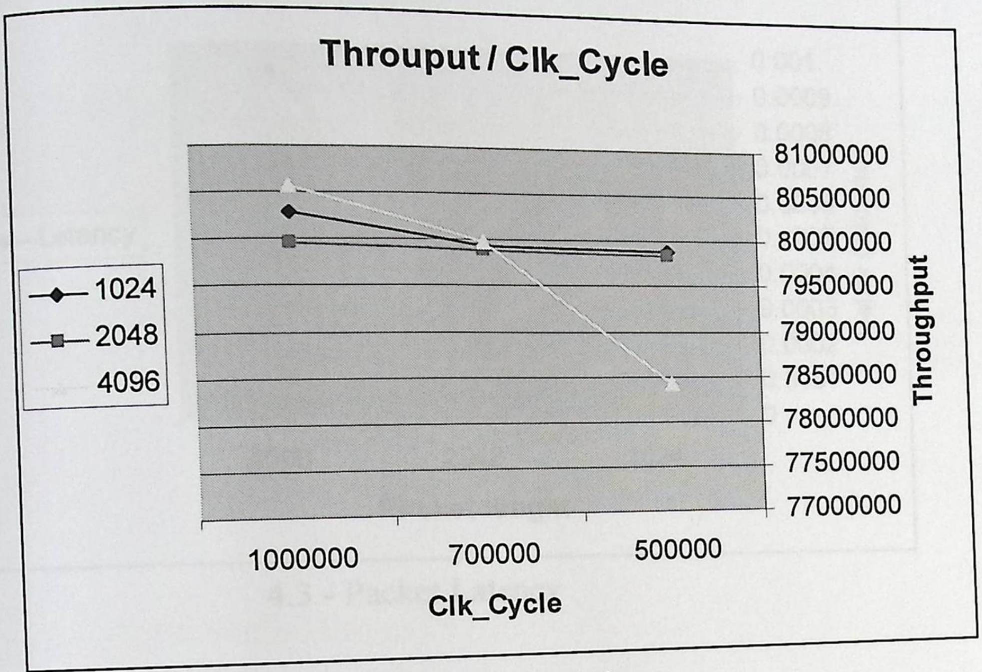
Table 4.1 – Bandwidth Result Table

And the figure below shoes the relationship between the Bandwidth and the packet length with different clk_cycle.



4.1- Switch Bandwidth

Figure 4.2 shows the saturation throughput for proposed network with varying packet length. It is clear from the figure that the proposed switch depends on the packet length but there is another factor effects on the result which is the wait time that each packet have to wait on the switch buffers



4.2- Switch Throughput

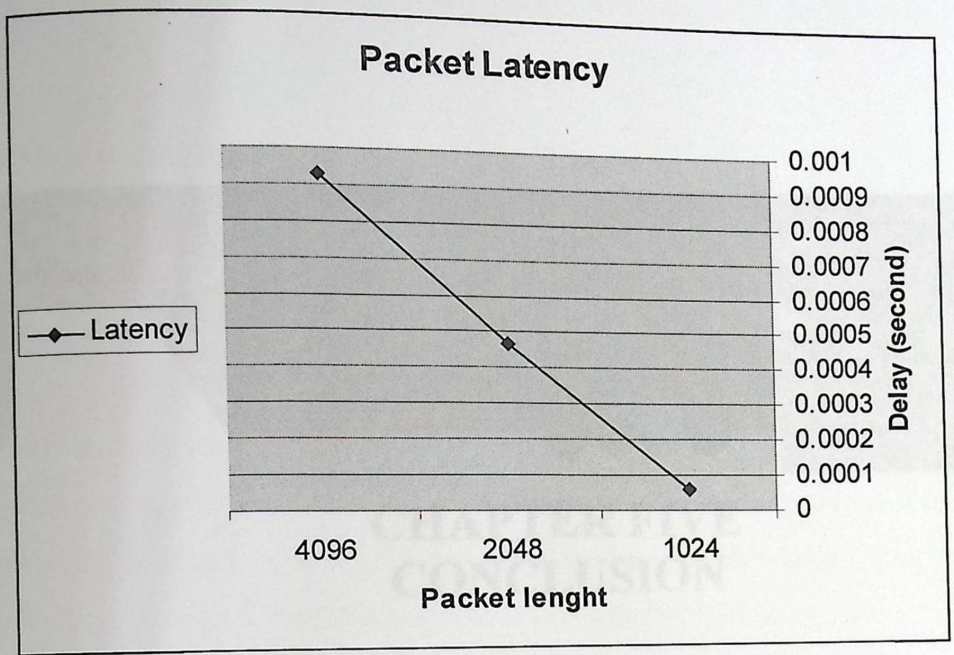
4.3 Packet Latency

The simulation process discussed in section 4.3 also produces results for latency viruses various values of packet length depending on table 3.1, 3.2, and latency equations in section. As discussed in chapter 2, 3 the network latency means how long it takes one bit to propagate from one end network to the other. Table 4.2 presents the simulator results.

Packet Length	Delay
1024	0.000062
2048	0.000465
4096	0.000929

Table 4.2 – Packet latency

In general the average latency increases with the packet length that means increasing in the packet length increases the latency which affects the bandwidth figure 4.3.



4.3 - Packet Latency

4.4 comparing the proposed network with other networks

This section presents the throughput and latency of proposed network with other networks figure 4.3 presents the relation between packet length and throughput for proposed network(31 nodes) and the Clos network(256 nodes).

It is clear from figure 4.3 that the Clos network is better than proposed network take in consideration that Clos network built using 64 nodes and the proposed network built using 31 nodes. Different number of nodes affects the networks load.

CHAPTER FIVE
CONCLUSION

5.1 Conclusion

5.2 Achievements

5.1Conclusion

In this work, a simulator for the proposed network was built to simulate its behavior under various traffic conditions. The simulator results were discussed in chapter four.

5.2 Achievements:

Through simulating the proposed network under various traffic conditions, the position of this network among other networks was defined to a bad degree of accuracy.

The performance (throughput, and latency) of this network was measured under various packet lengths to determine where best it can be used and compare it with other networks. Extensive simulation tests and measurements have shown the proposed network is not suite for use, because the throughput and latency are not reflecting the better results compared with other networks.

As a future work the simulator can be modified for a large number of nodes (switches and pc's), large packet length, and various loads.

We recommend to continuo this work until achieve the objectives.

APPENDIXS and REFERENCES

List of References

[Haas 98] Stefan Haas, “The IEEE 1355 Standard: Developments, Performance and Application in High Energy Physics”, PhD thesis, Liverpool University, 1998.

[Hwa 93] Kai Hwang, " Advanced Computer Architecture: Parallelism, Scalability, Programmability", McGraw-Hill Inc., N.Y, U.S.A,1993..

[NI 01] National Instruments," Matrix Switch Expansion Guide", National Instrument Corporation. June 2001, Application Note 174.

[shi 01] Hongyuan Shi and Harish Sethu, “Virtual Circuit Blocking Probabilities in an ATM Banyan Network with *bXb* Switching Elements”, (part of Advanced Simulation Technologies Conference) Seattle, Washington, USA, April 22–26, 2001

[Yang 99]Yuanyuan Yang, Gerald M. Masson: The Necessary Conditions for Clos-Type Nonblocking Multicast Networks. IEEE Trans. Computers 48(11): 1214-1227 (1999).