Copyright © 1987, by the author(s).
All rights reserved.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission.

LOW-COST HIGH-SPEED GENERAL SERVICE FIBER OPTIC NETWORKS

by

Ming-Kang Liu

Memorandum No. UCB/ERL M87/26

20 April 1987

COVER PAGE

LOW-COST HIGH-SPEED GENERAL SERVICE FIBER OPTIC NETWORKS

by

Ming-Kang Liu

Memorandum No. UCB/ERL M87/26

20 April 1987

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

TITLE PAGE

LOW-COST HIGH-SPEED GENERAL SERVICE FIBER OPTIC NETWORKS

by

Ming-Kang Liu

Memorandum No. UCB/ERL M87/26

20 April 1987

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Low-Cost High-Speed General Service Fiber Optic Networks

Ming-Kang Liu

ABSTRACT:

Fiber optics is an emerging low cost technology for high speed communications, and the information age has been characterized as a time of exploding communication demands of all kinds. In this thesis, system solutions to low cost but high performance implementation of fiber optic communication networks for general services are proposed.

A fiber optic network distinguishes itself from other networks by its large transmission bandwidth. As a result, it is electronics that limits the available bandwidth for the information exchange. Also, a general service network has the capability to support all video, voice, and data traffic. With quite different traffic characteristics, a general service network should be designed so that all messages are handled efficiently and satisfactorily. In the thesis, system solutions which (1) minimize the speed mismatch between optics and electronics and (2) optimize the service for different traffic are considered.

These solutions include: (1) a circuit switching approach called Time Slot Switching (TSS), (2) an integration of TSS and Carrier Sense Multiple Access (CSMA), (3) an integration of TSS and slotted-ring, (4) and a simple high speed timing recovery method suited for fiber optics. Circuit switching provides multiple simultaneous links; therefore, the throughput can be much larger than with packet broadcasting such as CSMA or token-passing. Its fixed transmission delay is also desirable for voice or video. Compared with packet switching, circuit switching avoids much of the high speed packet processing; consequently, the speed constraint from electronics is mitigated.

The integration of TSS with CSMA or slotted-ring provides better services for data traffic and for networks covering larger geographical areas. We show that this integration is attractive for better performance and simple implementation.

The new timing recovery approach for high speed fiber optic communications avoids the use of PLL or SAW filters. The basic idea is to use a code with special properties which dramatically simplify the timing recovery process. This has been experimentally verified at 200 Mb/s using ECL logic. This suggests that Gb/s speeds can be achieved in bipolar integrated circuits or a few hundred Mb/s in CMOS.

These results contribute to achieving high speed fiber optic networks with low cost and high performance.

David G Messerschmitt

David G. Messerschmitt

Committee Chairman

"The fear of the Lord is the beginning
of knowledge,
but fools despise wisdom and discipline."

Proverbs 1:7

To My Parents, *Yee-Tang and Pei-Yu Liu,*

for Their Endless Love

ACKNOWLEDGEMENTS

I am deeply indebted to my research advisor, Professor David G. Messerschmitt, for his intelligent guidance and persistent support throughout my study. I wish to express my gratitude to Professor David A. Hodges, who initiated the project and provided me with valuable technical knowledge and experimental environment. I would also like to extend my appreciation to my committee members: Professors Finbarr O'Sullivan, Jean Walrand, and Michael Lieberman.

Many other people made themselves available during my researches: Mr. Hyun J. Shin gave me very strong support with the high speed circuit switch he designed and implemented; Miss Janet M. Cooper in Bell Communications Research contributed important experimental work; Professor Robert G. Meyer and Mr. Mehmet Soyuer provided very valuable technical discussions; and Dr. Harry Mussman in AT&T Bell Laboratories offered indispensable optical fibers, transceivers, and wiring tools. The work cannot be completed without their help.

I would also like to thank my friendly colleagues: Graham Brand, Teresa Meng, Tom Chen, Biswa Ghosh, Vijay Madisetti, Keshab Parhi, Ilovich Mordechay, and etc., who constantly support me and give many suggestions. In particular, I should thank Tom Chen for carefully reading and commenting my draft.

This work was supported by grants from Advanced Micro Devices, Fairchild Semiconductor, Harris Semiconductor, National Semiconductor, Intel Corporation, and Bell Communications Research, with a matching grant from the University of California MICRO program.

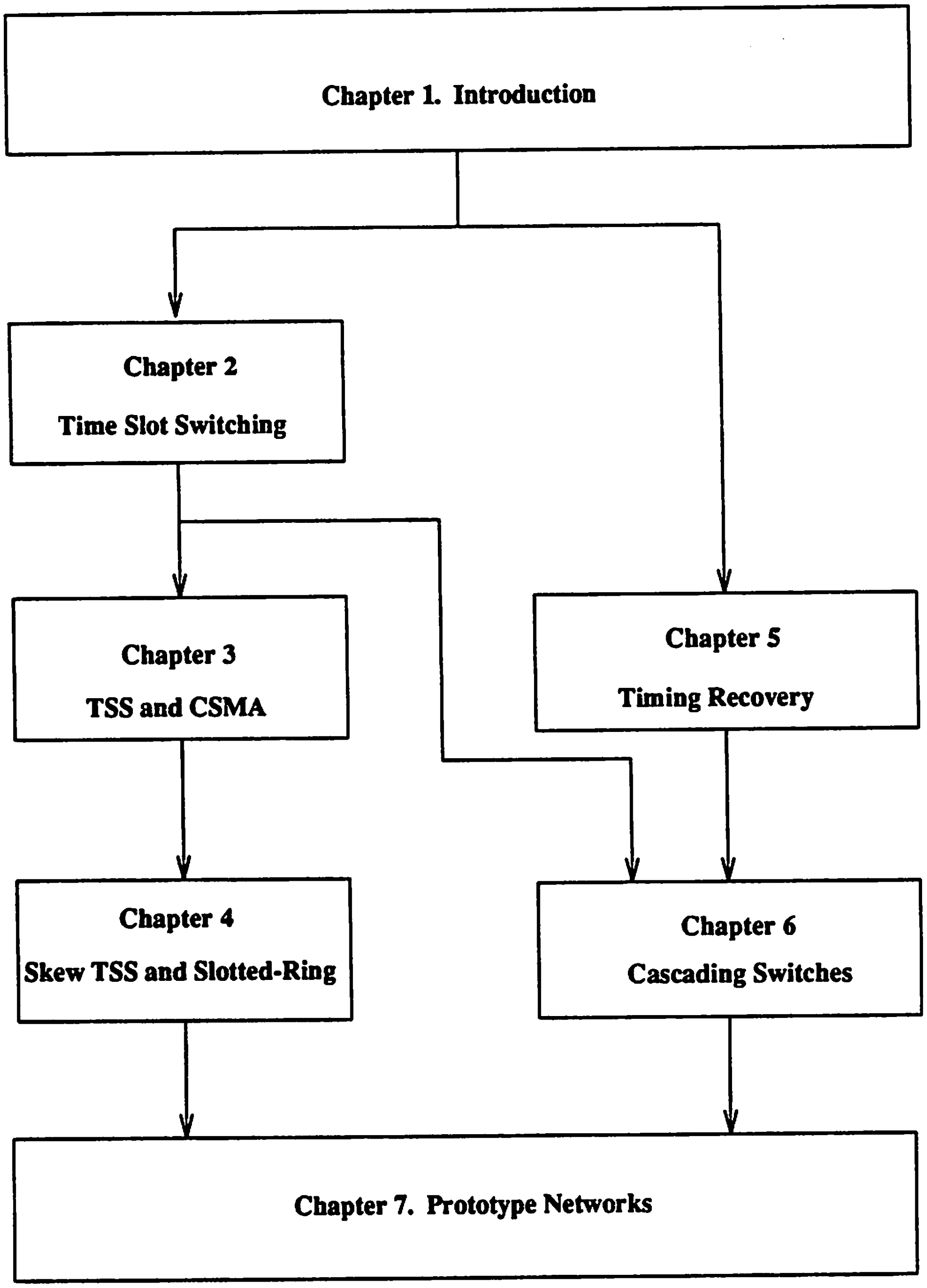
I should finally express my deep appreciation to my wife, Ai-Ching, who carefully arranges my life and accompanies me over countless nights.

Table of Contents

Part I: Asynchronous Time Division Networks

Part II: High Speed Network Implementation

The Relationship of the Chapters



CHAPTER 1

Introduction

1.1. Introduction

A communication network consists of nodes interconnected through some transmission medium by which a large number of users communicate with each other. A network can use radio, coaxial cable, pair wire, or fiber optics as its transmission medium, and it can span over a campus, a city, or even a whole nation. It can serve different functions: voice, data, video information, or some combination. In this thesis, a fiber optics network to serve various types of traffic will be studied. The network's range can span a local area to a larger metropolitan area.

This chapter explains the objective of this thesis, fundamental backgrounds, and a brief introduction to the approach. Detailed discussion will follow in the remaining chapters.

1.2. The Objective

Fiber optics provides low cost and high immunity to noise, and has become a good choice for the transmission medium in local area and metropolitan area networks [1-7]. Also, technologies of integrated electronics, especially high performance CMOS circuits, have improved significantly in both switching speeds and function complexity during last decades [8-10]. With these advanced optics and electronics, communication bandwidth has been greatly increased.

The "Information Age" has been characterized as a time of exploding communications demands of all kinds. These new demands, for example to provide a general service for all voice, data and video, create great challenges for system engineers. Our current communications technologies have been designed around specific applications and these different applications impose different constraints on the network (e.g., allowable transmission delay). The telephone network, cable networks, and data packet switching networks (e.g. the ARPA Net) are examples.

Interestingly, minicomputers, workstations and personal computers have gradually replaced the roles of main frames for data processing and computation. In response to this change, small area networks such as local area networks (LAN) have emerged to provide the intercommunications between these small computers [11]. Ethernet, Token-ring, and Slot-ring are successful examples of such networks [12-15]. In more recent years, information exchanges between LANs, PBXs, and main frames have also been needed. This resulted in the Metropolitan Area Network (MAN) concept [16-19].

In summary, a general purpose local and metro area network is getting more and more important, and as a result, the objective is *to take advantage of well developed and low cost electronics and optics technologies to serve voice, data, and video traffic in local or metropolitan areas.* There are three major problems to be solved:

(1) Traffic congestion in a network.

(2) Integrating the different service requirements of voice, data, and video traffic.

(3) The speed mismatch between electronics and optics.

These will be elaborated in the following sections.

Foundations to the approaches to solving these problems -- classifications of networks, characteristics of traffic, special features of LANs and MANs, and properties of fiber optics and integrated electronics -- are discussed first. Brief descriptions of the approaches will also be given before detailed discussion in the remaining chapters.

1.3. Network Classifications

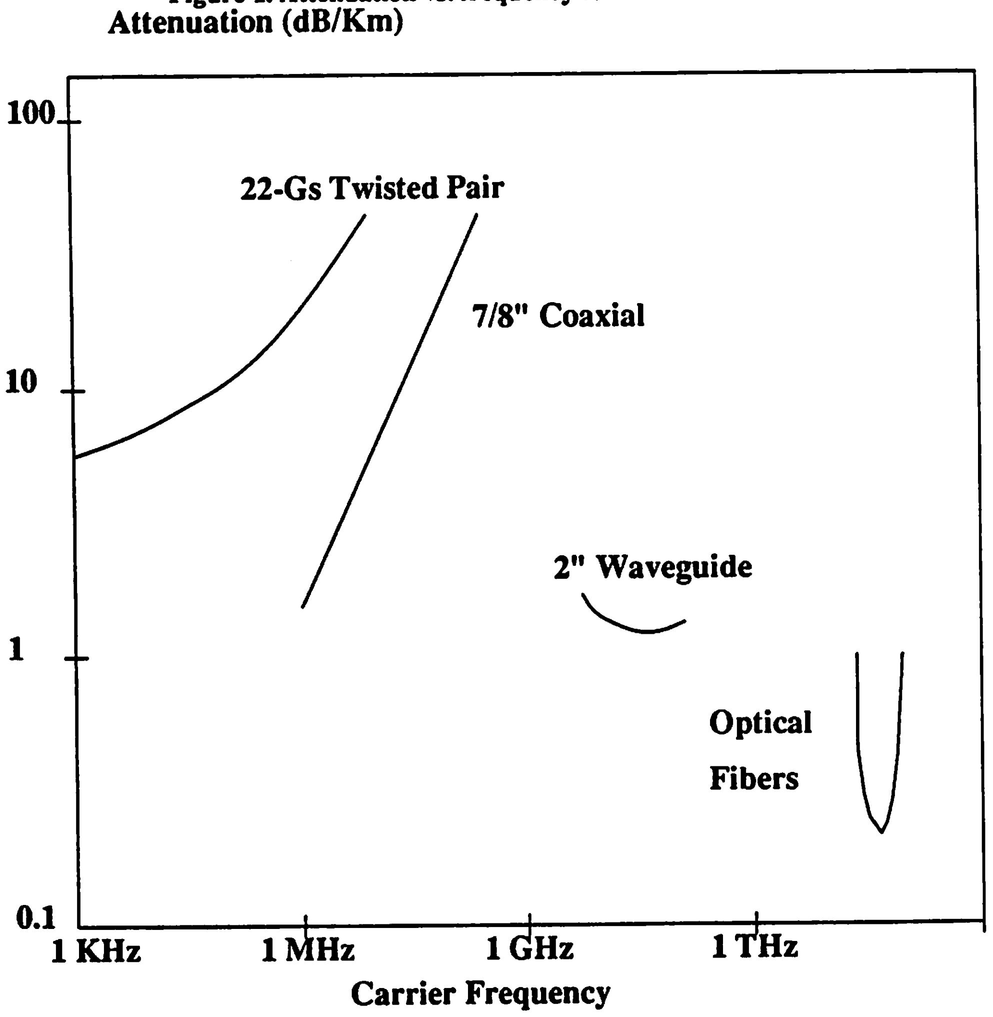
A network can be classified by its: (1) transmission medium, (2) size, (3) traffic types, (4) bandwidth, (5) topology, and (6) medium access protocol.

1.3.1. Transmission Medium

Electrical signals can pass through various kinds of media, with variations in attenuation, noise interference, and channel bandwidth. Examples are twisted pair in telephone subscriber loops, coaxial cables in cable TVs, free space for radios, and fiber optics for high bandwidth applications.

A good communication channel is characterized by low signal attenuation, low noise interference level, and large signal bandwidth. If analog transmission is used, linearity will also be important. A comparison of the attenuation as a function of carrier frequency among twisted pairs, coaxial cables, waveguides, and optical fibers is shown in Figure 1, [20]. In general, the bandwidth is proportional to the carrier frequency; as a result, optical fiber is the best medium for transmission in terms of the attenuation and bandwidth. Another important feature of optical fibers is its very low noise interference. Radio interference, multiple path effect, cross talk, and even strong signal jamming would have very little effect

Figure 1. Attenuation vs. frequency for different media



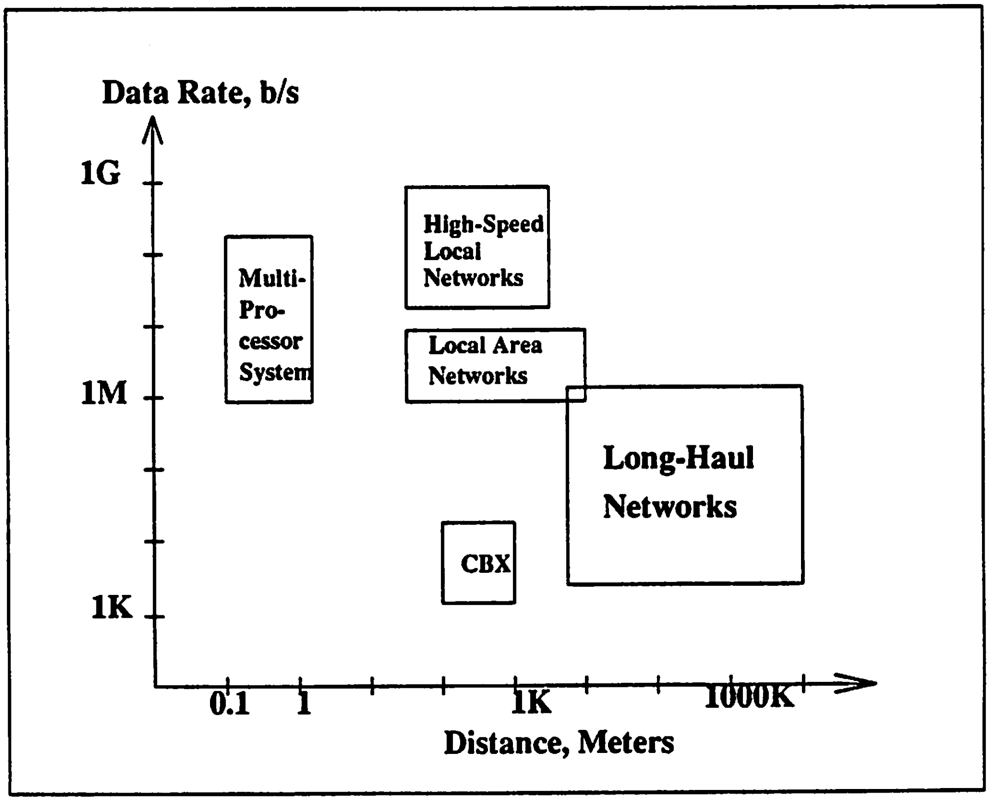
to the signals in fibers. This makes fiber optics more attractive than other media in many important respects.

1.3.2. Network Size

Not only the transmission medium but also the network size affects the quality of a communication channel. A larger size requires longer transmission lines and therefore a higher noise level, larger signal attenuation, and smaller bandwidth due to the dispersive effect. These facts make bandwidth over a long distance very expensive, and we can see in Figure 2 [21] that there is a tradeoff in the network size and bandwidth for the existing networks.

The network size also has a very important influence from the system point of view. A fundamental limitation is that the propagation delay is linearly proportional to the network size, and the larger the propa-

Figure 2. Data rate vs. network size



gation delay, the more difficult to synchronize all the users sharing a common medium [22]. For small area networks such as LANs, the propagation delay and signal distortion are both small. These leads simplicity in network designs possible. Section 4 will study this in detail and it will be clearer as we go thoroughly in next three chapters.

1.3.3. Network Bandwidth

The available network bandwidth strongly depends on the transmission medium and the network size, as just explained. Larger bandwidth networks can provide more general service (e.g. video and data) and lower queuing delay.

1.3.4. Network Topology

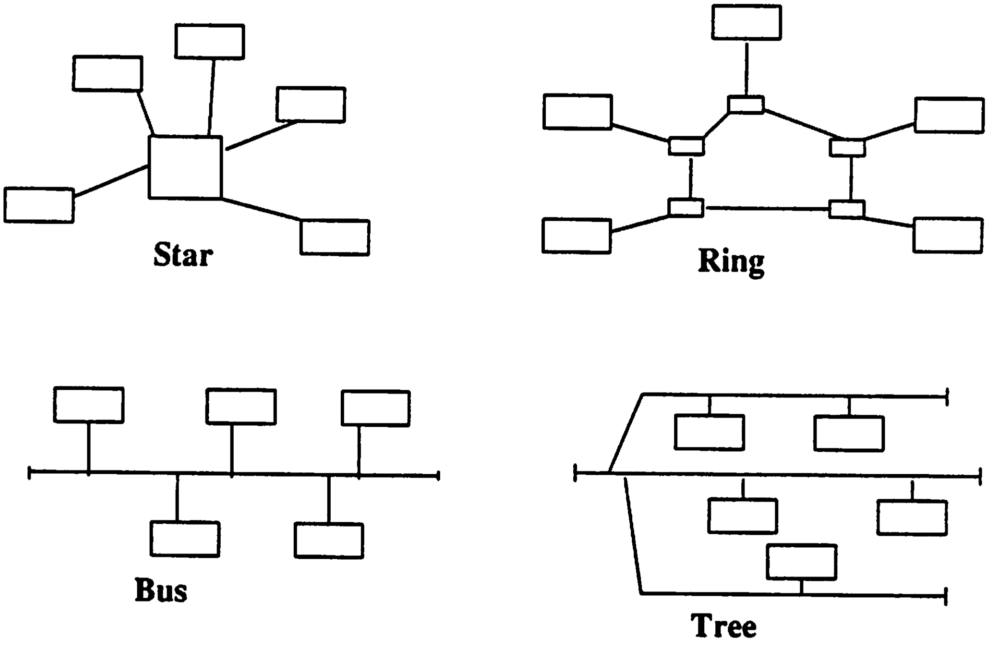
The network topology is the geometrical relationship of communication nodes in a network. The topology has strong correlations with its transmission medium, medium access protocol, system reliability and flexibility, and traffic throughput. Figure 3 shows some typical network topologies: star, ring, bus, and tree.

Generally speaking, the bus or ring has a linear structure and therefore traffic flow can be more regular; as a result, they are potentially more efficient in resolving access conflicts, especially when the network size is large. Token ring or token bus is an example. On the other hand, since the traffic has to go in one direction and through all the links, the total throughput is limited by the link bandwidth [22]. In addition, they have lower reliability because any node's failure may partition the network. The star provides more flexibility in traffic flow by using a circuit switch inside the central star. In addition, in fiber optics networks, the optical interface to the network prefers the star topology [7]. The disadvantage is that a central controller is required to manage the switching.

1.3.5. Traffic Types

Voice, data, and video have quite different characteristics in arrival statistics, bandwidth, delay tolerance, and error tolerance, and a medium access protocol should satisfy all their needs. Detailed discussion will be given in the next section.

Figure 3. Different network topologies



1.3.6. Medium Access Protocol

A medium access protocol defines the rules for users to share the common medium; in other words, it specifies how different traffics are multiplexed in a network. The first choice of multiplexing is in time or frequency domain. In this thesis, only time domain multiplexing will be considered, in which there is another choice: broadcasting or switching. Discussion of these two categories follows.

Broadcasting network techniques including token-ring, slot-ring, and carrier-sensing have been used to time-share a single physical channel among many stations for LANs. The IEEE 802 standard has established well defined protocols for LANs. The 802.3 specification is for Carrier-Sense Multiple-Access with Collision Detection (CSMA/CD) on a bus, while standards 802.4 and 802.5 are for token-passing on a bus and a ring, respectively [23-24]. The advantages of this approach are simple processing (no routing, no central controller, for examples) and low cost (no switching hardware and software required). But broadcasting has the disadvantage that total network throughput is limited by the bandwidth of a single communication link.

Switching allows multiple transmissions at the same time and therefore provides a higher throughput. There are two switching methods: packet or circuit. In packet switching, X.25 recommended by CCITT [25], for example, data is divided into packets, and involves packet capsulation and decapsulation, routing, sequencing, acknowledgement, and error recovery. For a fiber optics environment, these require high speed processors matched with the high transmission rate of optical links. On the other hand, circuit switching used in telephone networks, provides a dedicated link by prior call request, and therefore involves no complicated real-time processings. However, protocols involving circuit switching at high speed and using fiber optics have received less attention, in part because of the unavailability of wideband optical or electronic circuit switches.

1.4. Traffic Characteristics

A general communication network should accommodate all kinds of data, voice, and video traffic. Traffic characteristics which can be described by its arrival process, bandwidth, delay tolerance, and error tolerance are examined below.

1.4.1. Voice

Voice traffic in most cases is interactive and requires real-time transmission. This limits the transmission delay to be within tens of milliseconds. Longer delays will make listeners uncomfortable and even noisy. On the other hand, speech is less susceptible to interference, conversation echoes, lost packets, than data.

The standard PCM digital speech requires a bit rate of 64Kb/s. During a conversation, statistically an average of 40 to 50% of time is active in each direction [26]. There are many time compression and interpolation techniques [27-30] to increase the bandwidth efficiency, but inevitably they introduce extra hardware and degraded quality. Since fiber optics in LANs and MANs provides abundant bandwidth, standard PCM voice will be simpler to use and will probably result in a lower overall cost.

1.4.2. Data

Data traffic can be either interactive or send-and-wait. Interactive data (e.g. RS232) has similar characteristics to voice except that the active time percentage in each direction could be much lower. Send-and-wait data has a collection of bits to transmit, such as E-mails or file transfers. In general it is non-interactive and allows a large variation of transmission delay.

Both these types of data all require a low error rate. Messages in error generally have to be retransmitted. Their arrival process compared to speech is very bursty. These characteristics make packet switching is more favorable than circuit switching.

1.4.3. Video

Video traffic in most cases requires real time transmission, and is not interactive. Compared to voice, it has much larger bandwidth (50 to 140 Mb/s) and 100% bandwidth usage. As a result, circuit switching seems to be most promising for video.

1.5. Local Area and Metropolitan Area Networks

Local area networks (LANs) were developed first for data communication among workstations or minicomputers [11,12], and have short propagation delay and small signal distortion. These features provide extra opportunities in the protocol design. For example, CSMA and ALOHA [31-32] both bear the same concept of random access, but only CSMA has collision detection (CD) capability to abort a collided transmission immediately. In general, the larger the propagation delay, the higher the "asynchronousness" between users. This asynchronousness reveals itself in different forms in different networks. For CSMA/CD, it is the time to detect a collision; for token-ring, it is the time to pass the token; and for TDMA (Time Division Multiple Access) [33], it is the guard time between adjacent time slots. Fortunately, when a LAN is concerned, this asynchronousness is negligible and can be traded for simpler protocols or implementations.

Low signal distortion offers an additional opportunity in implementation. If a packet passes a few nodes which are not the destination, timing recovery would be unnecessary because timing distortion is not significant. For fiber optics with low noise, small dispersion, and little attenuation, no timing recovery is necessary, thereby reducing the cost in implementation.

Metropolitan area networks (MANs) have a larger network size (about 50 Km); therefore, the propagation delay becomes more significant than in LANs. Techniques in LANs consequently can not be applied directly and need at least some modifications. For example, CSMA/CD would be unfeasible in MANs because the overhead in collision time is very significant, and IEEE 802.6 [18] is modified from the slot-ring protocol to fit MAN applications.

On the other hand, if MAN is used for the intercommunication between LANs, PBXs, and main frames, the structure of a network for this purpose could be simpler than than that of LANs. This has an analogy that wire connections of central offices in telephone networks are much simpler than those of local user distributions. This important property will be utilized in the protocol studied in chapter 4.

1.6. Fiber Optics and Integrated Electronics

Over the past 15 years, successful efforts have been made to increase the bandwidth and reduce the energy attenuation in fibers. Now inexpensive optical fiber media capable of several Gb/s data rates over point-to-point links of several kilometers are available. It has become very attractive to use optical fiber as the transmission media for LANs or MANs.

To produce economical fiber optic systems, the electronic functions required in transmitters, receivers, multiplexors, circuit switches, buffers, etc. must be eventually integrated as VLSI components. Unfortunately, neither silicon nor gallium arsenide technology can provide VLSI implementations capable of operating at rates higher than a few hundred Mb/s. While this is adequate for integrated voice, data, and video services to a single station, it is not high enough to meet the needs in a broadcast network of tens or hundreds of stations.

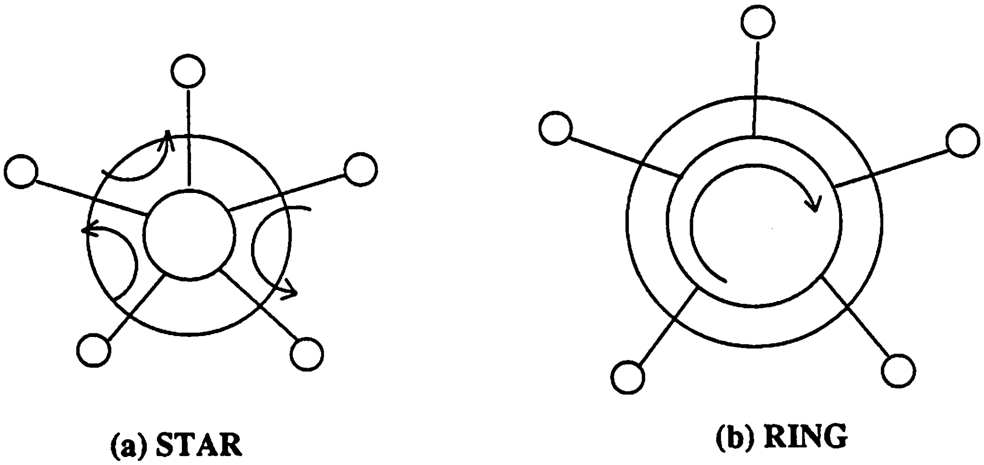
An additional impetus for introducing switching into a fiber optic network is that many years of research on optical devices have failed to produce practical three-way optical couplers that would be needed for networks based on a shared optical bus [7]. The reason is optical couplers have insertion loss which limits the total number of taps on a bus. Active couplers are subject to failure unreliability and high cost. In addition, practical optical switches that could be the basis for a switched optical network are not available. However, recent work has demonstrated the feasibility of economical electronic circuit switches in CMOS VLSI, operating at up to 200 Mb/s data rates per channel [34,35]. Assuming the availability of economical switching techniques, a network based on this requires an opto-electronic transceiver pair for every link at each switching node. However, avoiding the costly timing recovery function at the switching node(s) can significantly reduce costs. Provided that crosstalk and timing jitter in a fiber-optic local area network can be adequately controlled, we will describe an approach which avoids the need for timing recovery in the switching nodes.

1.7. The Approaches

Instead of going to more exotic and costly technologies to achieve a higher rate network utilizing a broadcast protocol such as CSMA or a token-ring, our approach will be to incorporate switching in the network, in order to gain almost unlimited bandwidth capability while at the same time running each link in the network at the highest speed consistent with low-cost technologies. A comparison between circuit switching and broadcasting architectures is illustrated in Fig. (4a) and (4b). Switching allows multiple transmission paths to coexist simultaneously, resulting in greater network throughput for a given link bandwidth. This approach offers in addition the ability to add network capacity incrementally, and it is inherently more reliable since failures result in incremental loss of capacity.

In considering switching, we have chosen circuit switching primarily because it allows a switching fabric that does not actually examine or process the bit streams passing through (as just explained in section 1.2.6), with the result that the bit rate can be considerably greater for a given low-cost technology. This greater speed can more than make up for the inefficiency in circuit switching data traffic with fluctuating bit rates, particularly in view of that fact that packet switching data networks can easily be overlaid on the circuit network using lower speed circuits as packet links. Furthermore, the modest efficiencies gained in packet switching voice and video do not appear to be justified in a small-area network where bandwidth is plentiful.

Figure 4. Circuit switching vs. broadcasting


(a) STAR

(b) RING

There has been previous work on LANs using circuit switching at 1 Mb/s [36] and 380 Mb/s [37]. With circuit switching, low speed traffic, like 64 Kb/s voice channels, needs to be multiplexed into high speed media. This can result in timing recovery and multiplexing functions which reduce the achievable bit rate. An additional problem with circuit switching is that when the number of switchable links is small, significant blocking will occur at a low level of network utilization.

These problems with circuit switching lead us to propose a variant which we call time slot switching (TSS). TSS can be described as a time-divided space-division switch in a particular form well-suited to LANs based on fiber optics. A detailed description of TSS is in chapter 2.

The electronic switch can not only be operated to switch different traffics, but can also be configured in a broadcasting way. This provides the flexibility in combining TSS with other broadcasting protocols for better serving some data traffic. Chapter 3 considers the integration of TSS and CSMA, and chapter 4 evaluates the combination of TSS and Slot-Ring. Their performance in LANs and MANs will also be studied in these chapters.

In chapter 5, an approach leading to the simplification of timing recovery is also considered. High speed timing recovery is essential but difficult to implement in digital communications. A fixed coding technique is proposed and tested. Finally, chapter 6 discusses tests of the effects of no timing recovery but just amplification when bit streams pass through many circuit switches connected by fiber optics by TSS as proposed in chapter 2.

REFERENCE:

[1] Morton I. Schwartz, *Optical Fiber Transmission - From Conception to Prominence in 20 Years,* IEEE Communication Magazine, May 1984, pp. 38-48.

[2] Eric G. Rawson, *Application of Fiber Optics to Local Networks* Proceedings of the LACN Symposium, May 1979.

[3] A. Husain and S. D. Cook, *Impact of Fiber Optics on Local Area Networking* SPIE Vol. 355, 1982, pp. 120-126.

[4] Tadashi Akiba, Akihiro Okada, Shun Suzuki and Osamu Takahashi, *Optical Bus Network Using Ethernet and Higher Level Protocol LSIs* Proceeding of ICC, 1984, pp. 1335-1339.

[5] S. D. Personick, *A Review of Alternative Fiber Optic Distributed Access Network Designs, - And The Problems They May Solve* Proceeding of ICC, 1984, pp. 898-902.

[6] Ronald V. Schmidt and Eric G. Rawson, *Fibernet II: a Fiber Optic Local Area Network with Data Collision Sensing,* SPIE Vol. 355, 1982, pp. 127-131.

[7] Marion R. Finley Jr., *Optical Fibers in Local Area Network,* IEEE Communication Magazine, Aug. 1984, pp. 22-35.

[8] J. A. Becker and J. N. Shive *The Transistor - A New Semiconductor Amplifier* Electrical Engineering, March 1949, pp. 215-221 and reprinted in Proceedings of the IEEE, Dec. 1984, pp. 1696-1703.

[9] Rick D. Davies *The Case for CMOS,* IEEE Spectrum, October 1983, pp. 26-32.

[10] Edward T. Lewis *Design and Performance of "1.25-µm" CMOS for Digital Applications*

[11] V. David Tsao, *A LAN Architecture Overview,* IEEE Communication Magazine, August 1984, pp.

7-11.

[12] Robert M. Metcalfe and David R. Boggs, *Distributed Packet Switching for Local Computer Networks,* **Communications of ACM, July 1976, pp. 395-404.**

[13] J. D. Markov and N. C. Strole, *Token-Ring Local Area Networks: A Perspective,* **Proceedings of COMPCON F82, pp. 606-614.**

[14] D. W. Andrews and G. D. Schultz, *A Token-Ring Architecture for Local Area Networks: An Update,* **ibid, pp. 615-624.**

[15] J. H. Saltzer, D. D. Clark, and K. T. Pogran, *Why a Ring?* **Proceedings of the Seventh Data communications Symposium, 1981, pp. 211-217.**

[16] Robert W. Klessig, *Overview of Metropolitan Area Networks,* **IEEE Communications Magazine, January 1986, pp. 9-15.**

[17] Floyd E. Ross, *FDDI - A Tutorial,* **IEEE Communications Magazine, May 1986, pp. 10-17.**

[18] Draft of Proposed IEEE 802.6 Standard, *Metropolitan Area Network (MAN) Media Access Control,* **August 1985, Revision D.**

[19] Daniel T. W. Sze, *A Metropolitan Area Network* **IEEE Journal on Selected Areas In Communications, November 1985, pp. 815-824.**

[20] Paul S. Henry, *Introduction to Lightwave Transmission,* **IEEE Communications Magazine, May 1985, pp. 12-16.**

[21] William Stallings, *Local Networks, An Introduction,* **ISBN 0-02-415460-1, 1984, Chapter 1.**

[22] C. Petitpierre, *Influence of the Network Topology Upon the Throughput Limitations,* **Proceedings of ICC 1984, pp. 428-433.**

[23] IEEE Project 802, *Local Area Network Traffic Handling Characteristics Committee Report*, working draft, January 1982.

[24] IEEE 802.3, IEEE 802.4, and IEEE 802.5, *ANSI/IEEE Standards, Draft International Standard, IEEE Inc., 1985.*

[25] CCITT Recommendations X.1, X.2, X.25, X.92, and X.96 *Public Data Networks*, CCITT Orange Book, Vol. VIII.2, Geneva, Switzerland: ITU, 1977

[26] Paul T. Brady, *A Statistical Analysis of On-Off Patterns in 16 Conversations*, The Bell System Technical Journal, January 1968, pp. 73-91.

[27] K. Bullington and J. M. Fraser, *Engineering Aspects of TASI* ibid, March 1959, pp. 353-364.

[28] J. M. Fraser, D. B. Bullock, and N. G. Long, *Over-All Characteristics of a TASI System*, ibid, July 1962, pp. 1439-1454.

[29] H. Miedema and M. G. Schachtman, *TASI Quality - Effect of Speech Detectors and Interpolation*, ibid, July 1962, pp. 1455-1473.

[30] Richard V. Cox and Ronald E. Crochiere, *Multiple User Variable Rate Coding for TASI and Packet Transmission Systems* IEEE Transactions on Communications, March 1980, pp. 334-344.

[31] N. Abramson, *The ALOHA System - Another Alternative for Computer Communications*, AFIPS Conference Proceedings, Vol. 37, 1970, pp. 281-285.

[32] R. Binder, et. al., *ALOHA Packet Broadcasting - A Retrospect*, AFIPS Conference Proceedings, Vol. 44, 1975, pp. 201-215.

[33] Kemilo Feher, *Digital Communications, Satellite/Earth Station Engineering*, Prentice Hall Press, 1981.

[34] Gary A. Hayward, A. M. Gottlieb, D. G. Boyer, and J. E. Berthold, *High-Speed 16x16 CMOS Crosspoint Switch, Electronics Letters, Vol. 21*, No. 20, 1985, pp. 923-925.

[35] H. J. Shin and D. A. Hodges, *CMOS Switch for 200 Mb/s Fiber Optic Networks*, to be published.

[36] B. Hailpern, A. Heller, L. W. Hoevel, and Y. J. Thefaine, *ALAN: A (Circuit-Switched) Local Area Network*, IEEE Jr. on Selected Areas in Communications, May 1985, pp. 427-430.

[37] Ulrich Killat and Johann Kruger, *System Aspects and Realization of Wide-Band Switching in the Local Area*, ibid, March 1985, pp. 330-335.

Part I.

Asynchronous Time Division Networks

CHAPTER 2

Time Slot Switching

2.1. Introduction

A new medium access protocol called *Time Slot Switching* (TSS) for use in fiber optics LAN is explained in this chapter. This protocol incorporates features of time division, space division, and time compression for users to share a common medium. With these features, data, voice, and video traffic can all be served in a single local area network. In addition to fiber optics used for transmission, VLSI CMOS electrical crosspoints are used to switch traffic within individual time slots. Based on these techniques, high network traffic capacity and low implementation costs can be achieved.

Although TSS can serve various kinds of traffic, operation of TSS for a specific type of traffic with one constant rate (homogeneous TSS) is first described in section 2. CMOS circuit switches in TSS in general are distributed in a network and controlled by a central controller which assigns time slots and crosspoints for requesting traffic. This traffic control management will be explained in section 3. To examine the performance, an analysis is presented in section 4 to show the trade-offs among traffic capacity, frame guard times, blocking probability for new circuit requests, and transmission delay. The analysis is done for 64 and 16 Kb/s channels, and the results show that TSS is more attractive than broadcast protocols for voice traffic or constant rate data traffic. In section 5, TSS is generalized for serving voice, data, and video traffic.

2.2. Homogeneous TSS Networks

By a "homogeneous" TSS network, we mean one which supports a multiplicity of circuits, all at the same constant rate (e.g., 64 Kb/s PCM voice channel); this homogeneous assumption will be relaxed in Section 5.

2.2.1. Network Topology and Architecture

A TSS network generally consists of switching nodes connected by fiber optics. The switching nodes connect simultaneous circuits by their internal space-division switches. Users access the switching nodes through concentrators called time compression multiplexers (TCM), which multiplex user traffic in a time-division fashion, figures (1,2). Detailed operations of the switches and TCM will be addressed shortly.

Figure (1a) illustrates a mesh network structure with two types of switching nodes. Type A switches are only connected with other switching nodes. Type B switches connect TCMs with Type A and/or other type B nodes.

Figure (1b) shows a type A switching node, drawn to illustrate link level (control) and physical level (data path) elements. The circuit switches need connection information from the central controller to set up communication links; this information may be conveyed by a separate (lower speed) network, or by the TSS network itself where dedicated slots are reserved for this purpose (see section 3).

Figure 1a. TSS network topology and architecture

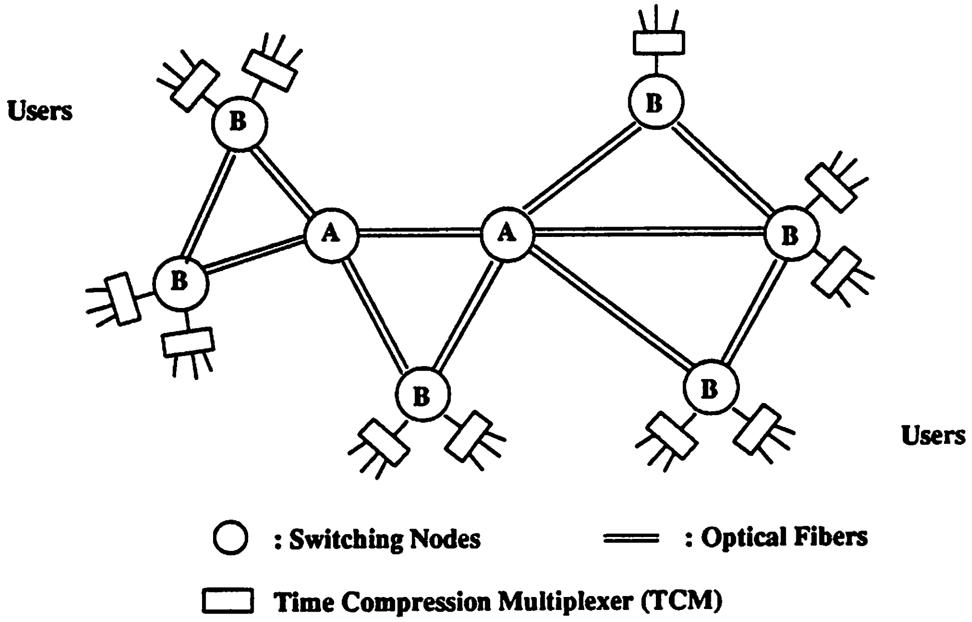


Figure 1b. Internal architecture of class A nodes

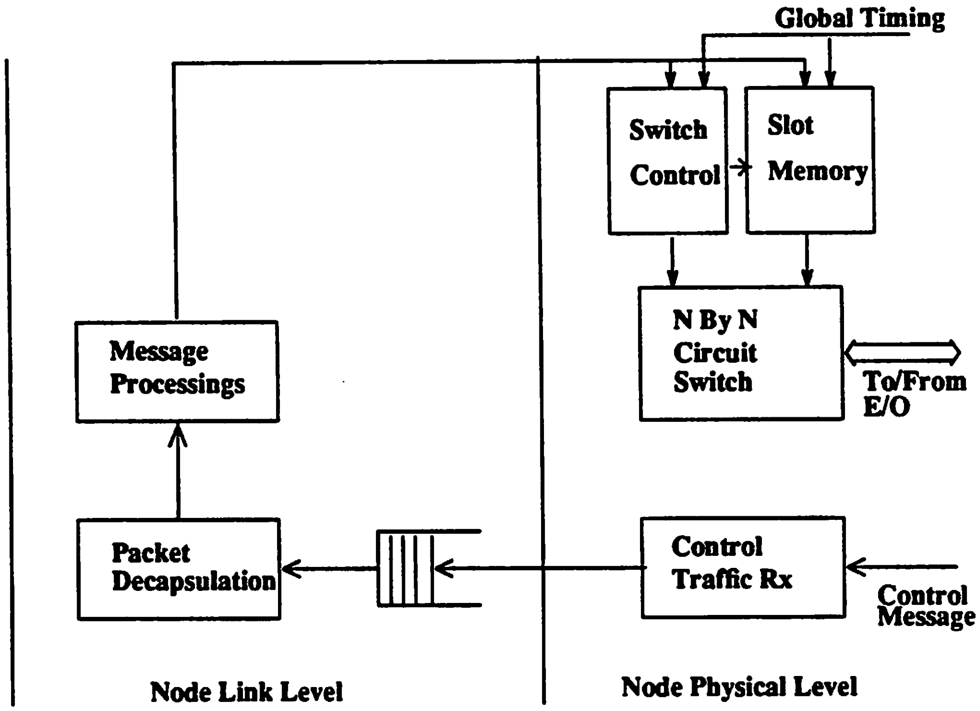


Figure 1c. Internal architecture of class B nodes

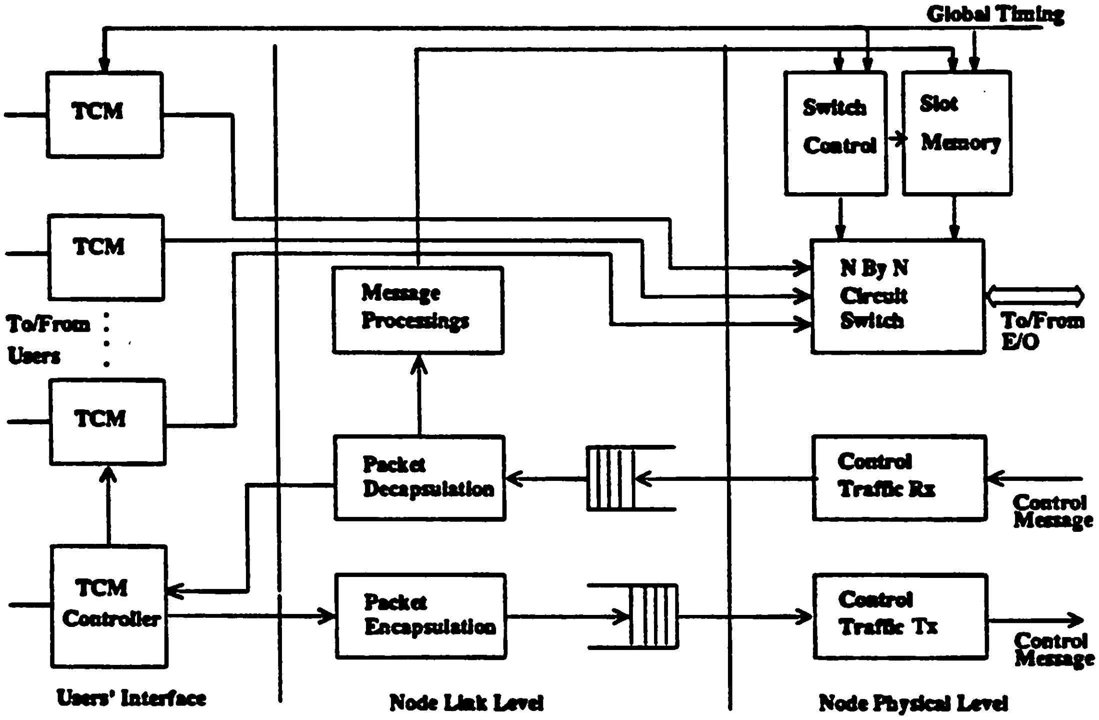
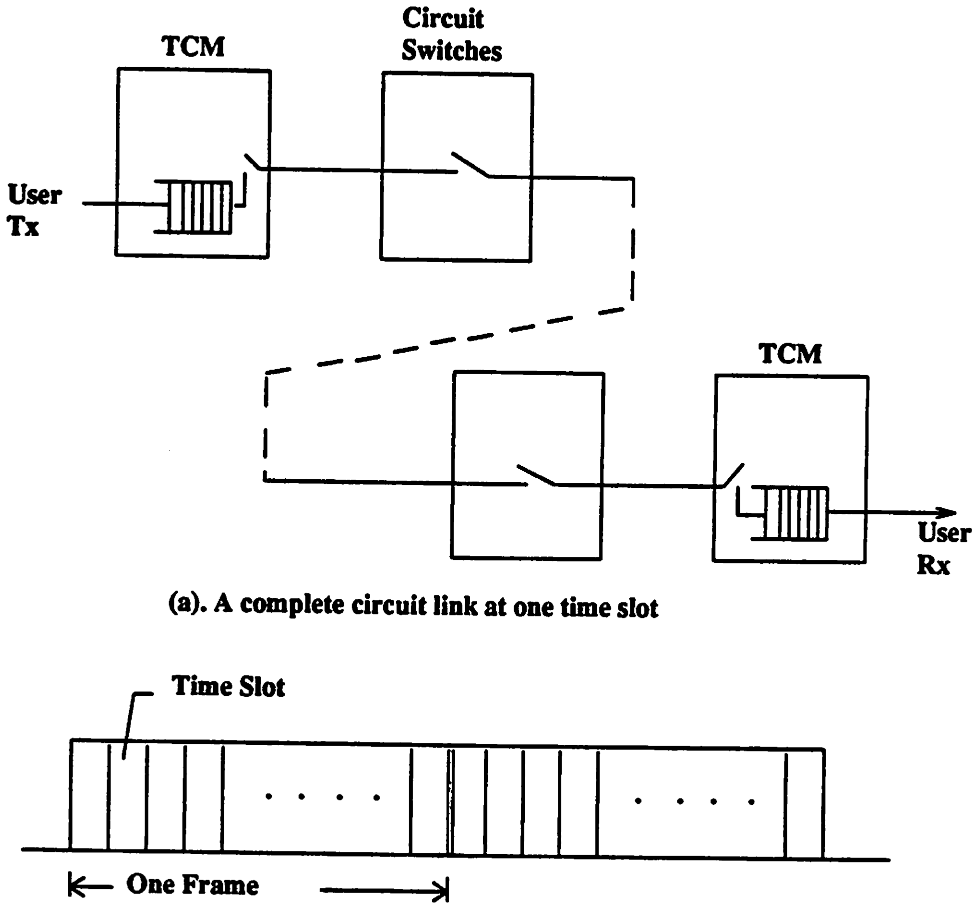


Figure (1c) shows a type B switching node, where it not only receives the connection information, but also transmits control messages like "circuit request" or "circuit finish" from the user terminals.

2.2.2. Basic Operations

Time slot switching (TSS) can be described as a switched time-division multiple-access (TDMA) network [1]. TDMA is often used in satellite networks for the same reason it is used here; namely, it allows very simple hardware for running the network at the maximum speed for a given technology. TCMs are used in TSS to give users TDMA access. In addition to the TCMs, TSS incorporates circuit switches to increase traffic throughput; in other words, the configuration of all the space-division switches

Figure 2. Basic circuit connections



(a). A complete circuit link at one time slot

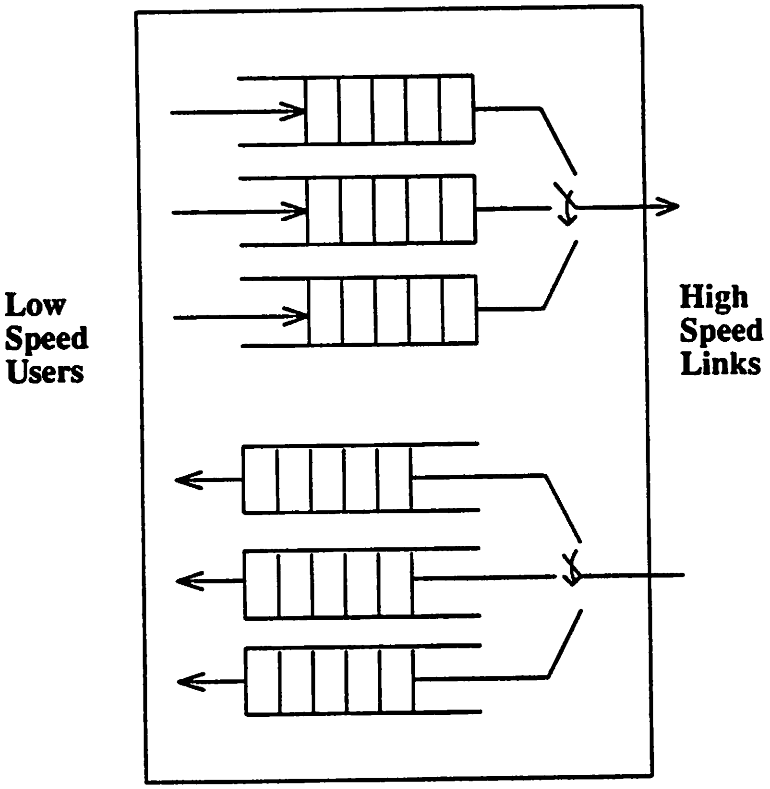
(b). Time Division Multiple Access

within the network changes for each time slot, thereby providing a multiplicity of circuits.

To set up a communication link, both the TCM and switches should operate synchronously in each time slot, figure (2); that is, a bit stream from a transmitting user is first multiplexed through a TCM to a circuit switch in a certain time slot, then arrives at the destination TCM through a correct connection of cross-points, and is finally demultiplexed to the receiving user at the same time slot. This synchronization between the TCM and switches is established by a pre-arrangement and will be discussed in section 3.

The TCM is an interface between low speed users and high speed circuit switches. The TCM performs two functions. One is to buffer a number of bits during one frame for lower speed users. Second, the TCM transmits or receives these bits within the appropriate time slot for which the desired connectivity has been pre-arranged through the space-division network. Each circuit for a particular user terminal will continue to access the same time slot in each frame for the duration of the circuit. Figure (3) shows the

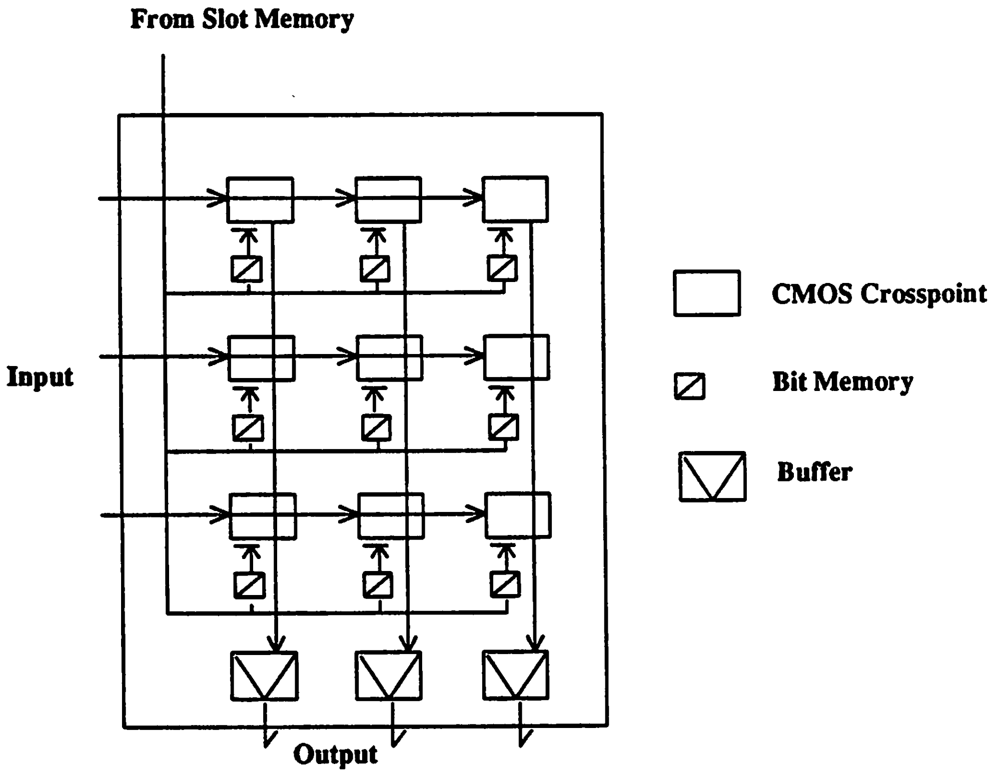
Figure 3. Internal structure of a TCM



configuration of a TCM. Here we see the user data streams are first stored in the transmitting buffers of the TCM. Typically more than one circuit will be established within a given TCM, and each circuit has its own buffer. Then during the selected time slots, the data in the corresponding buffers is transmitted over the high speed link. Similarly, on the receiving side, data arriving on a high speed link within pre-selected time slots are stored in receiving buffers, and then transmitted at a continuous slower speed to the user terminals. To summarize, a lower speed circuit is *compressed* and transmitted over the higher speed channel in a specific time slot in the frame, and at the receiver the circuit is *decompressed* for that same time slot to the lower speed user terminal.

The Circuit Switches connect physical paths for different communication links. An internal architecture of a circuit switch consists of an electrical cross-point matrix, a bank of slot memory, and control

Figure 4. Internal structure of a circuit switch



3 By 3 Circuit Switch

circuitry, figure (4) [2]. The slot memory contains connection information used for changing the configuration of the cross-point matrix after each time slot. The control circuitry provides slot timing and updates the connection information. In each time slot, there are simultaneous circuits being transmitted by the switches, each at a very high speed (e.g 200 Mb/s) compared to that of users. These multiple links combine to a total traffic capacity much greater than that achieved by broadcasting protocols such as token passing or CSMA operating at the same link speed.

2.2.3. Some Remarks on TSS

For the foreseeable future, electronics rather than optics is the factor that limits the bandwidth of practical fiber optic networks. TSS is primarily motivated by the desire to minimize the impact of electronic limitations:

1. The traffic passes through the TSS network *asynchronously;* in other words, no examination or processing of bit streams is required except at the source and destination nodes, where the data can be handled in a parallel format up to the final parallel-to-serial conversion in the TCM. This minimizes high speed electronics constraints.

2. Because the asynchronousness of the traffic flow, different instantaneous bit rates can coexist within the same network. For example, the logically separate control network can operate at a slower speed within a dedicated time slot, eliminating the necessity for using the same high speed electronics technology in the implementation of the control as compared to the switching function itself.

3. Timing recovery is required only once for each link, in the destination node (this is similar to broadcast networks). High-speed timing recovery is a costly function in opto-electronic transceivers; this function would be needed for every channel through every switch if bit or byte synchronization were required. In contrast, the function of low-speed timing recovery for frame synchronization is needed only once per switch node (section 3).

4. Collisions are eliminated by the arbitration provided by a central controller.

5. Circuit switching results in a deterministic delay, which is desirable in voice traffic and some data communication.

TSS also has disadvantages of course:

1. Though the switching nodes receive the same global timing, they are distributed in the network. Timing skew among these switching nodes is inevitable. This, coupled with the propagation delay of signals through the network results in the need for guard times in each time slot, and a resultant reduction in traffic capacity. This is very similar to the situation of satellite communication by TDMA with distributed earth stations. The need for reasonable throughput efficiency limits the geographical size of the network, although this limitation can be circumvented by using gateways with internal buffering.

2. Circuit switching results in blocking [3]. At a given traffic intensity, by increasing the total number of time slots in a frame, the blocking probability can be reduced, but at the expense of a larger frame size, which results in a larger transmission delay. The frame size can not be arbitrary large (see Eq. (9)); however, within its possible range, an achievable traffic intensity level can be estimated by specifying the blocking probability (e.g. 10^{-3}), Eq. (7).

3. TSS results in a buffering delay, called the compression time, on the order of the time duration of one frame. This is due to the need in the TCM to store one frame of data for transmission at a higher link bandwidth. This presents a problem in voice and interactive applications. (The first three disadvantages are analyzed in detail in section 4.)

4. Intermediate space-division switches without retiming introduce timing jitter due to dispersion on the fiber and crosstalk within the switch. This will limit the number of intermediate switching nodes and hence the allowable network topologies. Experiments to quantify this limit has been carried (chapter 6), and the results showed that the timing jitter is not noticeable.

5. The basic circuit switching architecture suggests a central controller. This, in addition to the requirement for global time slot timing, introduce possible points of vulnerability to single-point failure. This can be overcome by careful design and redundancy in these functions.

2.3. Switching Management

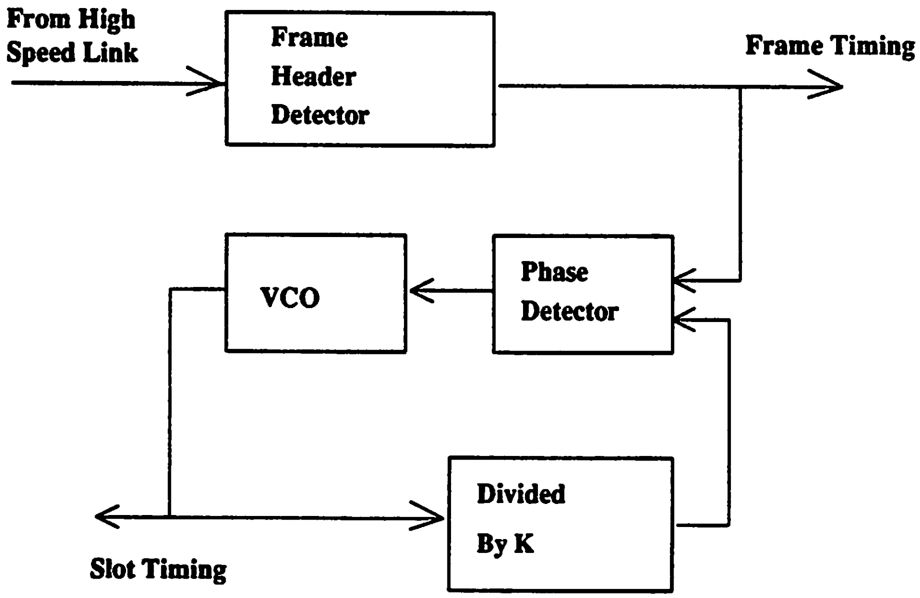
Because TSS has both time and space division switching, the TCM and switches need (1) frame timing, (2) slot timing, (3) cross-point connection information, and (4) TCM multiplexing information to set up correct communication links.

2.3.1. Frame and Slot Timing

The frame and slot timing are needed to change the TCM and switch configurations. These configurations are different in each time slot, and repeat each frame.

The frame timing can be obtained simply by generating a frame header of a special pattern and frequency from the central controller, just as in TDMA in satellite communications [1]. In TSS, this frame header can be put at the beginning of each frame; and by sensing this particular pattern, the frame timing can be extracted. Since both the frame and slot timing is at relatively low frequency (e.g. 20 Hz if each frame is of 50 msec, and 20 KHz if each slot is of 50 µsec), a simple PLL and a frequency divider can gen-

Figure 5. Frame and slot timing recovery



(K is the no. of slots in a frame)

erate slot timing synchronized with respect to the frame timing, figure (5).

2.3.2. Circuit Connection Setup

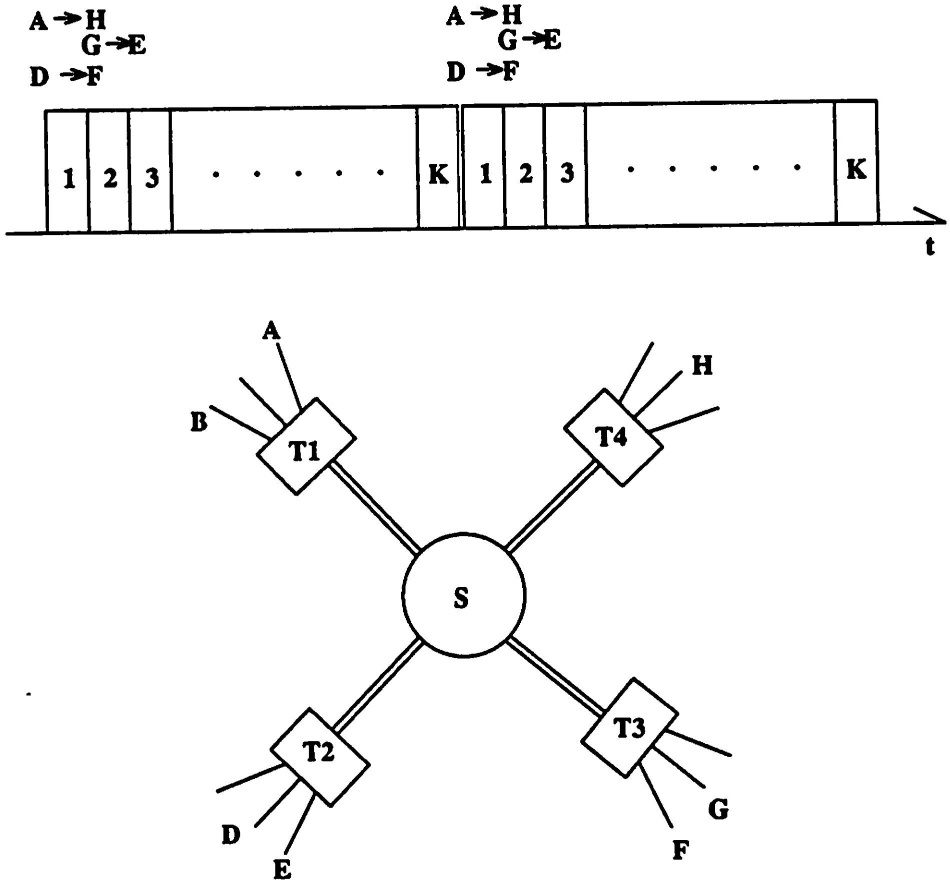
The switches and TCM need connection information to set up circuits correctly in each time slot. Since the switching nodes are generally distributed in the network, this connection is determined and distributed by the central controller after the controller receives "call requests" or "call finishes" from users. This subsection describes an algorithm for the controller to set up circuits, and the next subsection suggests how to pass the connection information between the switches and controller.

First, the central controller has a cross-point allocation table for each circuit switch and each time slot. After receiving a circuit request from one switching node to another node, the controller linearly searches for the first available time slot from the beginning of the frame. Generally, the circuit path may not be unique for a call, switches and cross-points of the minimum distance path will be chosen if there is no particular reason. Other searching algorithms such as random search are also possible, but will not be covered in this chapter.

As an example illustrated in figure (6), user terminal A is currently transmitting to user H during time slot 1 of each frame. TCMs T1, T4 and the switch S are synchronized to provide this link. Now, user terminal D wants to transmit to user F. There is no other traffic through TCM T2 and T3, and the current link between A and H causes no conflict in using circuit switch S for this new request. Thus the central controller can assign any slot for this new request, and it chooses slot no. 1 by the algorithm There will be two independent circuits in the same time slot. A few moments later, user terminal G wants to communicate with user E and sends the request to the central controller. There is no conflict with the existing circuits in the central circuit switch S, but T2 and T3 are reserved for D to F at slot no. 1. Consequently, the central controller will assign time slot 2 for this new request.

If there is neither an available time slot nor an available space-division path for a new requested circuit even the destination is idle, blocking occurs. The probability of blocking is estimated in the next section. Any circuit can be terminated by transmitting an appropriate "disconnect" message to the central controller, releasing the associated time slot for future use.

Figure 6. A single star network and its circuit setup procedures



2.3.3. The Control Traffic

A logically separate signaling network is needed to provide the connection information as mentioned above. This traffic demands can be estimated as follows: suppose there are 1000 users with active probability 50% (i.e. 50% of users are most likely using circuits at any time), and an average of circuit holding time is three minuntes such as in voice conversations, there will be only $1000\cdot\frac{0.5}{180}=2.8$ calls/min. Suppose each circuit establishment requires 1 Kb/call (to send source and destination addresses to the controller, and to send the connection information to the switching nodes), then the total control traffic demand is only about 3 Kb/s.

As mentioned before, though this control traffic can be supported by a separate lower speed network, it can be supported by the same TSS network with the first few slots reserved. For example, in the network shown in figure (1), let the switching node of type A on the right be the central controller also, the connection information for all the switches and TCM can be broadcasted at the second time slot of each frame, and the call request and finish information from all the switching nodes of type B can be sent to the controller at the third and forth time slots (at least two slots needed since there is a circuit conflict of two type B switching nodes on the left). It is noted that the first slot is already assigned for the extraction of the frame timing.

2.4. Performance Analysis of Homogeneous TSS

In this section, we present a quantitative analysis of the behavior of the TSS network. For simplicity, we treat the homogeneous case in which all circuits have the same bit rate. In order to quantify the impact of the choice of the bit rate, we consider two cases: 16 and 64 Kb/s.

In the analysis of TSS, we must consider a number of dependent parameters: the number of time slots per frame, the length of a time slot, the blocking probability for new circuit connection requests, utilization, and the compression delay. In the remainder of this section, we determine the following:

1. The guard time, required to protect the data in adjacent slots. It will be shown to be proportional to the maximum propagation delay in the network.
2. The blocking probability, which is a function of the topology and available circuits. We approximate this probability for a single or double star network topology.
3. The compression delay, which depends on the total number of time slots per frame, the slot size, and hence the utilization.
4. The preceding allows us finally to study the tradeoff between network utilization and the compression delay for a fixed blocking probability and network topology.

Before proceeding with the analysis, some definitions are appropriate:

Network Circuits: The average number of circuits connected at one time. Using telephone terminology, this has the units of Erlangs, where one Erlang is equivalent to one circuit continuously connected.

Network Utilization: The ratio of average total network throughput (bits per second) to the capacity of one link. Since multiple links are utilized in TSS due to switching, the network utilization in general can be greater 100%. This definition of utilization enables us to compare the traffic capacity to that of a multiple access protocol using a single shared link with the same capacity.

Blocking Probability: The probability that a circuit initiation request will be denied due to the absence of an available circuit path for any time slot between source and destination.

Compression Delay: The delay introduced in the most efficient implementation of TSS is equal to the time duration of one frame, which we call the compression delay. The delay in a practical implementation is likely to be slightly larger than this, since the propagation delay will be comparably smaller.

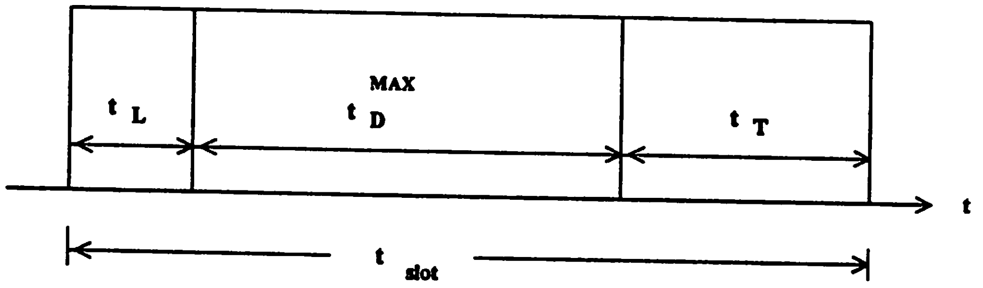
2.4.1. Guard Time Analysis

A time slot shown in figure (2.b) is shown in more detail in figure (7), where t_L is the leading guard time of a time slot, t_T is the trailing guard time of a time slot, and t_D^{MAX} is the interval of time available for actual data transmission. Then we have an expression for the interval of time corresponding to a time slot,

$$t_{slot} = t_L + t_D^{MAX} + t_T \equiv t_D^{MAX} + t_{idle} \qquad (1)$$

The requirement for non-zero t_L and t_T derives from finite propagation delay and timing skew (due to distributed global timing propagation delay between the central clock and each switching node in the net-

Figure 7. Internal structure of a time slot



work).

The slot utilization η_{slot}, defined as the fraction of the slot bandwidth actually used for data transmission, is

$$\eta_{slot} = \frac{t_D^{MAX}}{t_{slot}} \tag{2}$$

From Eq.(2), we want to minimize the guard times t_L and t_T. This minimal condition of $t_L + t_T \equiv t_{idle}$ can be obtained as in figure (8). Starting with t_L, the purpose of the leading guard time is to insure that a packet does not arrive at a node prior to the start of a time slot. Suppose node A sends a packet to node B where the global time slot timing of Node A leads that of Node B, as shown in figure (8a). To ensure that the beginning of the packet from Node A does not arrive too early, we have

$$t_L + t_{pg,AB} \geq t_{\Delta,AB}$$

where $t_{pg,AB}$ is the propagation delay between nodes A and B, and $t_{\Delta,AB}$ is the timing skew for the time slot clock between nodes A and B.

Similarly, the purpose of the trailing guard time is to insure that the end of a packet from node B to node A occurs before the end of the time slot as shown in figure 8b. The requirement is that

$$t_T \geq t_{pg,AB} + t_{\Delta AB}$$

The above conditions have to hold for any two switching nodes, so we have

$$t_L = \begin{cases} 0, & \text{if } t_{\Delta,\max} < t_{pg,\min} \\ t_{\Delta,\max} - t_{pg,\min}\ , & \text{otherwise.} \end{cases} \tag{3}$$

Also,

$$t_T \geq t_{\Delta,\max} + t_{pg,\max} \tag{4}$$

The worst case occurs when $t_{pg,\min} = 0$ and $t_{\Delta,\max} \approx t_{pg,\max} = t_{pg}$, in which case

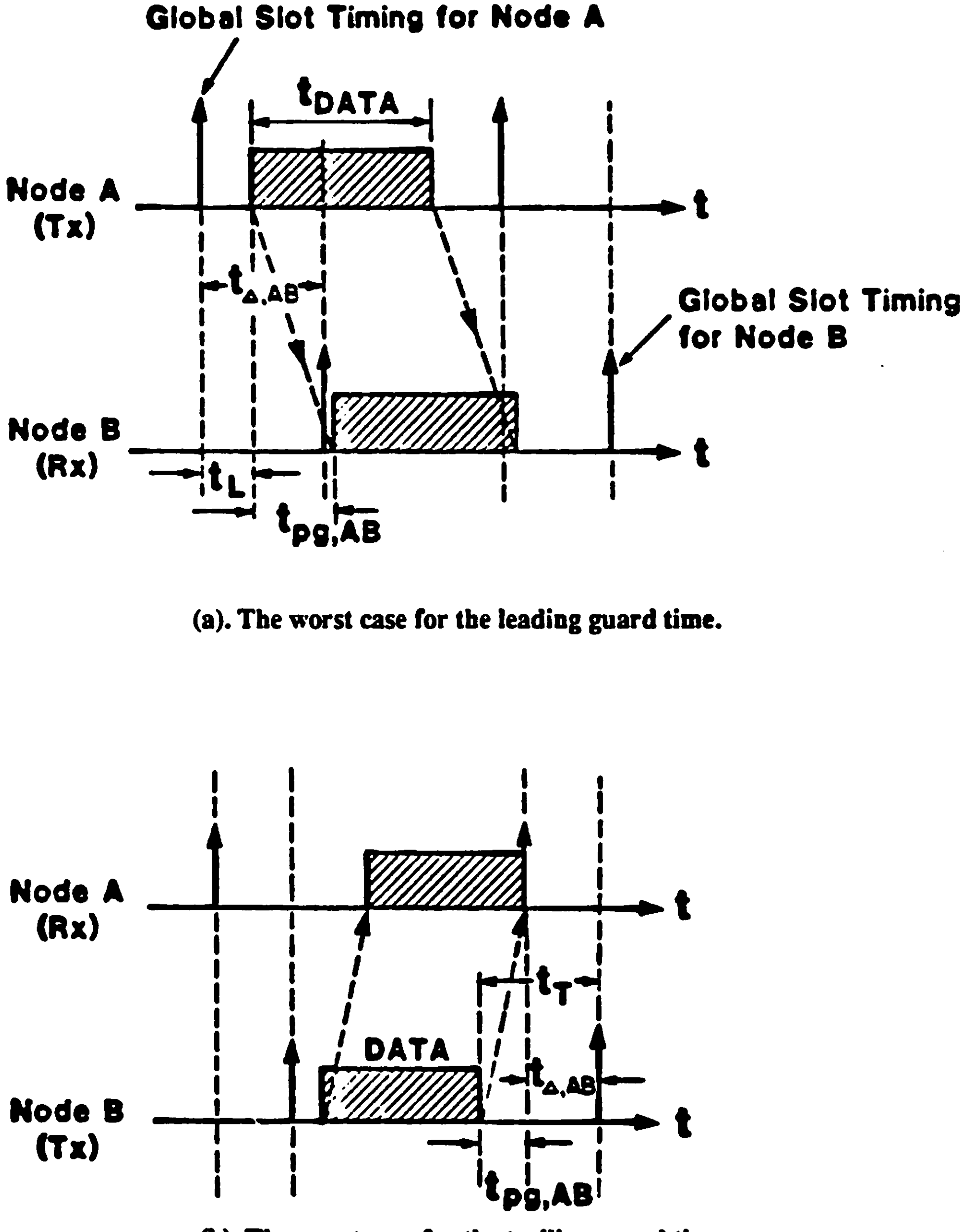
$$t_L = t_{pg}$$

$$t_T \approx 2\,t_{pg}$$

$$t_{idle} \equiv t_L + t_T \approx 3t_{pg} \tag{5}$$

Because t_{pg}, the maximum propagation time throught the network, cannot be reduced without compromising the size of the network, the only way to minimize the guard time t_{idle} is to minimize the

Figure 8. Guard time analysis for the worst cases



(a). The worst case for the leading guard time.

(b). The worst case for the trailing guard time.

timing skew $t_{\Delta,max}$. If we borrow a technique from TDMA satellite communications, where guard time also exists because of distributed earth stations [1], the guard time can be reduced by estimating the distance between the earth station and the satellite in the initialization of the switching nodes. Similarly, the propagation delay between a switching node to the central controller in TSS can be estimated, and the frame timing can be offset by this amount. That is, the time skew effect in the guard time can be reduced to a minimum; in that case, by Eqs. (3-4), t_L will be 0 and t_T will be only one t_{pg}. The idle time could be reduced to

$$t_{idle} \approx t_{pg}. \qquad (6)$$

In particular, if there is only one switching node in TSS, just like the case where many earth stations communicate to each other by only one satellite, the propagation effect in the trailing guard time (Eq. (4)) will also be zero, and the guard time can be made to approach to zero.

2.4.2. Network Circuits and Blocking Probability

In this section, we calculate the relationship between the average number of active circuits and the blocking probability in TSS. For a given network topology, any TSS network can be transformed into a topologically equivalent space-division switching network for which the blocking probability can be estimated by the method of Lee [3]. To illustrate, a simple single star network in figure (6) is considered. There are M TCMs connected to the central M by M switch, K time slots in each frame of the TCM, and $N < K$ user terminals connected to each TCM. As shown in figure (9), this switching network is equivalent to a space-division network with three stages. There are K (M by M) switches in the middle stage, one for each time slot, corresponding to the single physical time-divided space-division M by M switch. Each TCM allows the N user terminals to access one of K time slots, and hence is topologically equivalent to an N by K space-division switch.

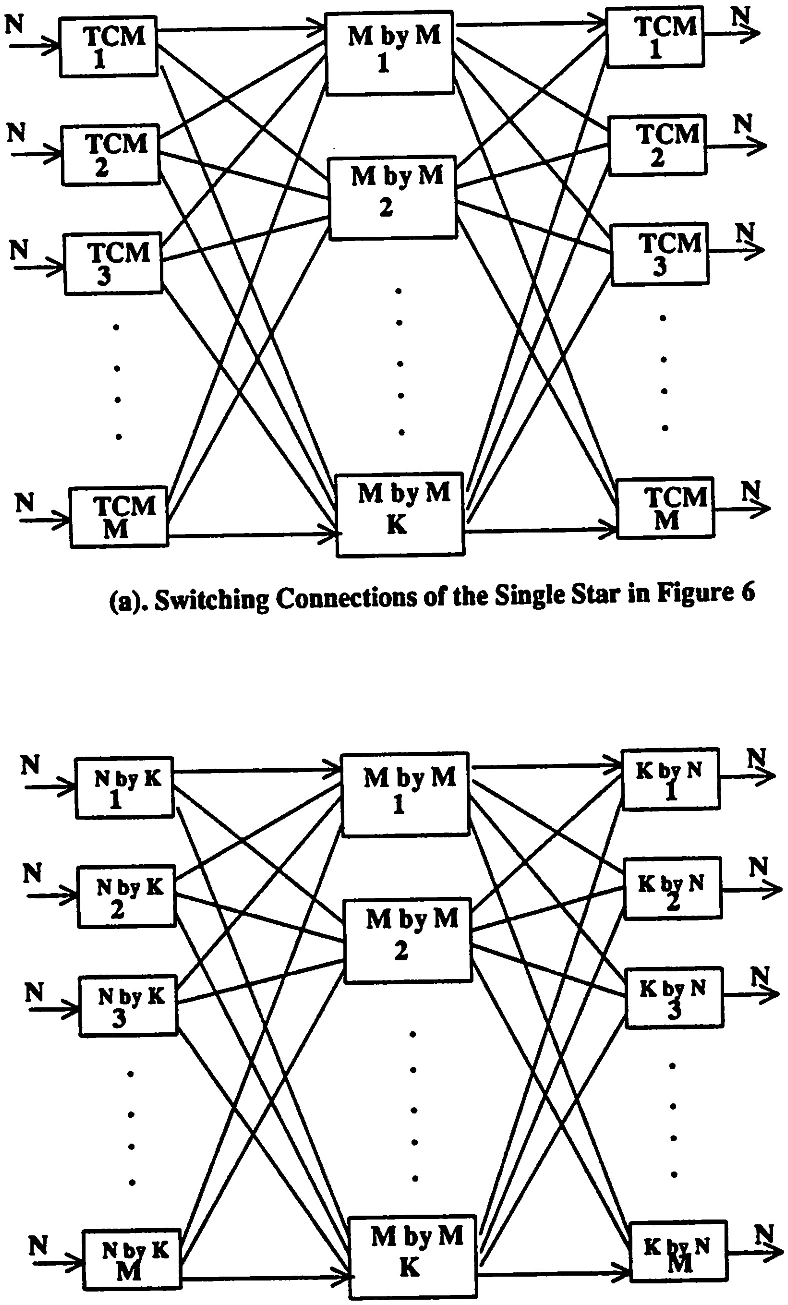
In general, except for special designs [4], switching networks have a non-zero blocking probability. Given a circuit switch structure, Lee [3] developed an approximate method of evaluating the blocking probability. For a single-star network shown in figure (6), by Lee's method (see Appendix A), we have the following approximation to the blocking probability,

$$P_{bk} = [1 - (1-p)^2]^K$$

$$p = \frac{A}{MK},$$

where A is the total offered traffic in Erlangs.

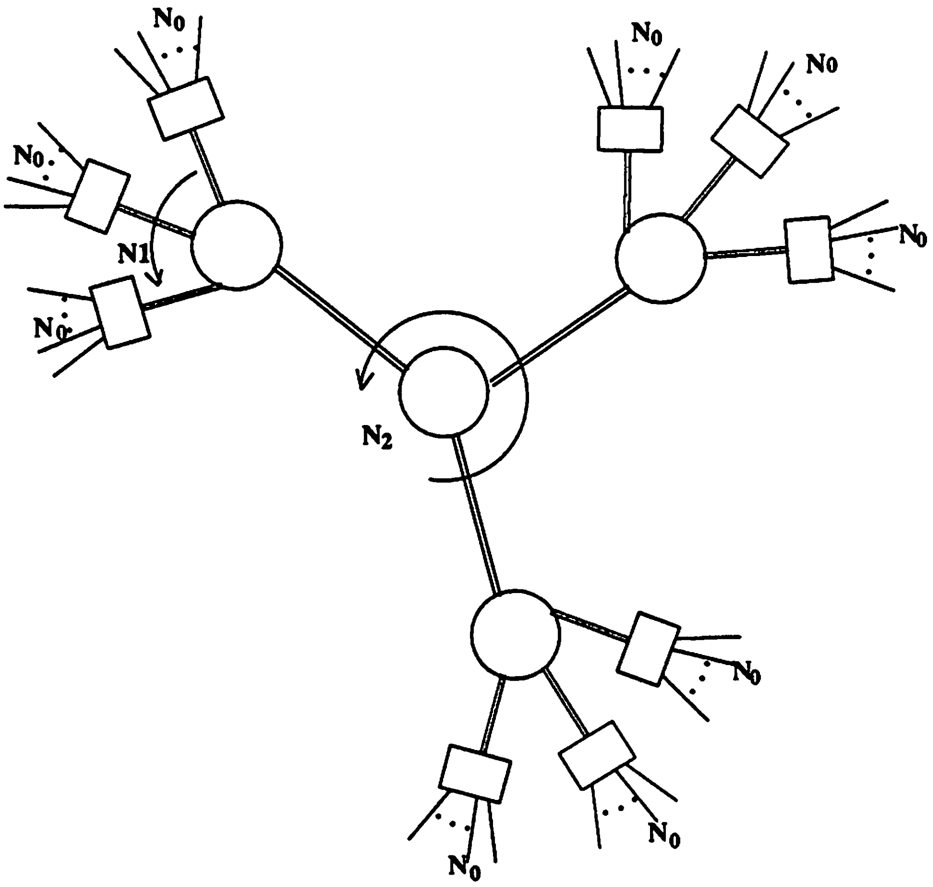
Figure 9. Logical connection comparison between a single star TSS and a 3 stage circuit switching network



(a). Switching Connections of the Single Star in Figure 6

(b). Equivalent Three Stage Space-Division Switching Network

Figure 10. A double star TSS network



A similar analysis for a a double-star network (Fig. (10), see Appendix B) gives

$$A \approx KN_2 \left\{ 1 - (1 - P_{bk}^{\frac{1}{2}})^{\frac{1}{2}} \right\}. \tag{7}$$

Figure (11) illustrates this relation numerically for a blocking probability of 10^{-3}.

Our experience is that Lee's method gives a conservative estimate of blocking probability; that is, it estimates a blocking probability higher than the actual. This is substantiated by simulation, the results of which are compared to Eq.(7) in figures (12) and (13). Lee's method of Eq. (7) will be used in the following performance analysis.

Figure 11. Achievable network circuits as a function of the total number of time slots in a frame and the size of circuit switches for the topology in Figure 10. The blocking probability is set to 0.001.

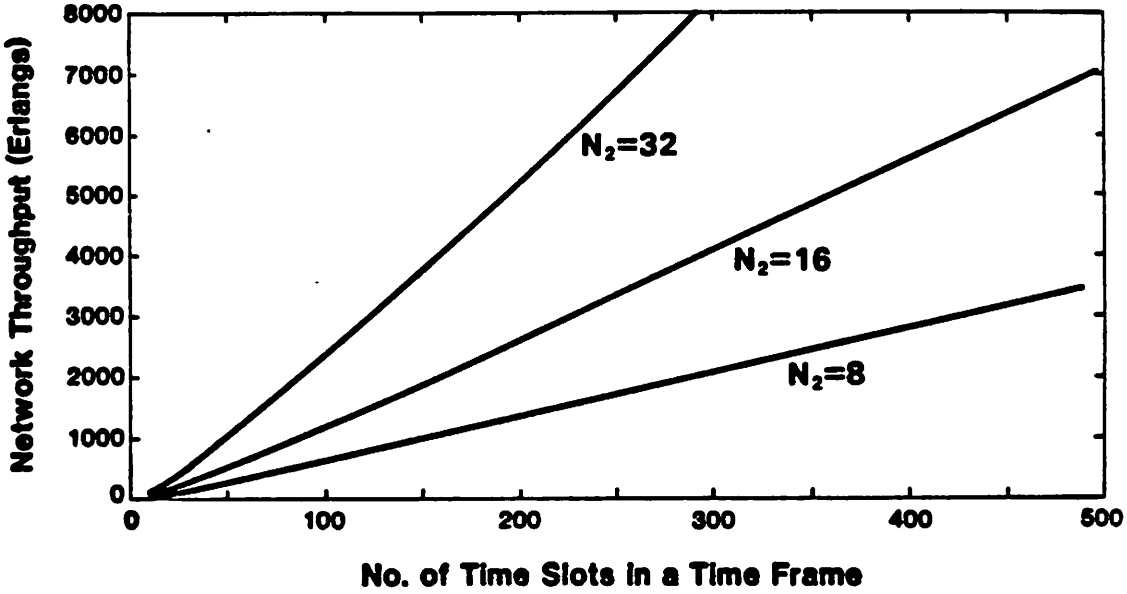


Figure 12. Blocking probability comparisons between the simulation and Lee's method, for the single star topology in Figure 6.

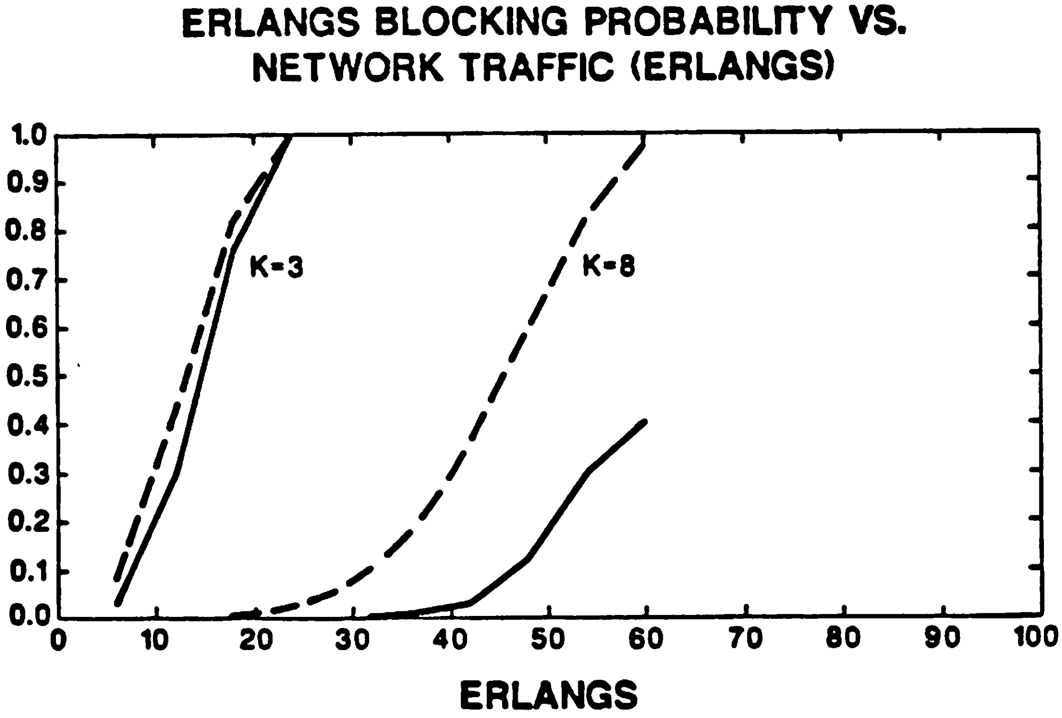
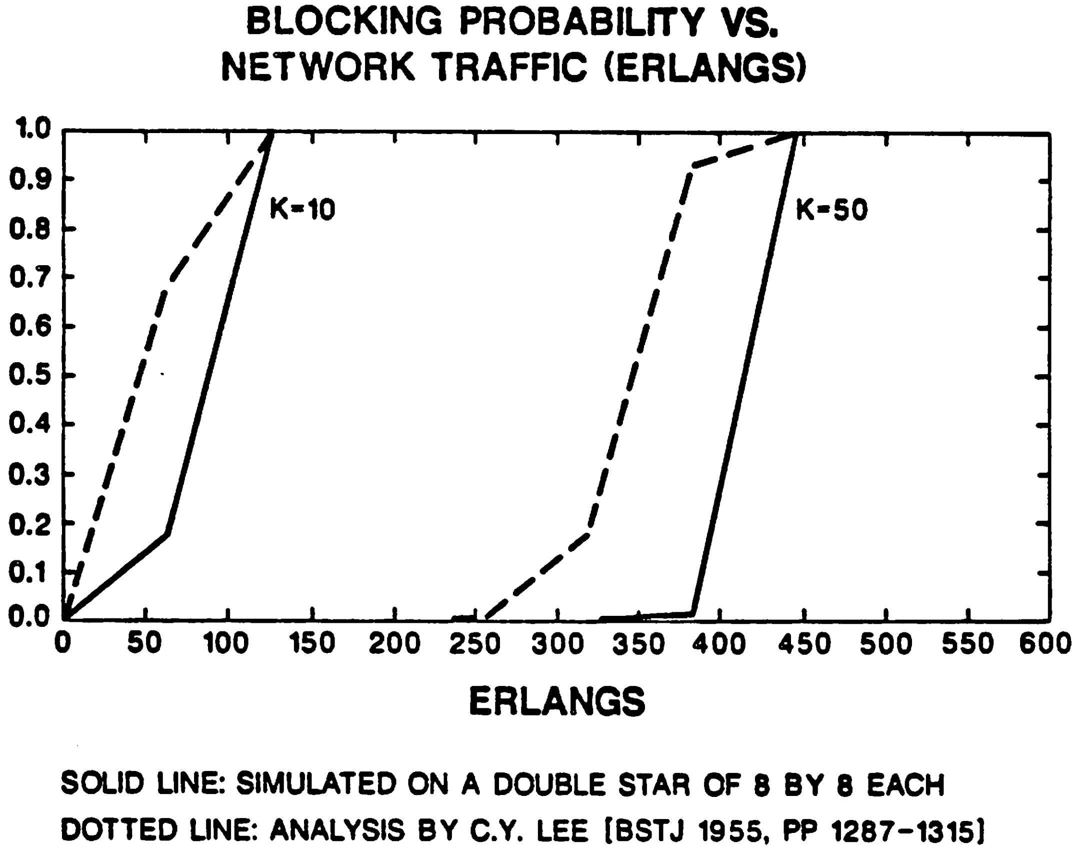


Figure 13. Blocking probability comparisons between the simulation and Lee's method, for the double topology in Figure 10.



2.4.3. Compression Delay

The compression delay in TSS is equal to the duration of one frame. Since there are K time slots in each frame, the compression delay is

$$D = K t_{slot}. \tag{8}$$

Considering now one circuit corresponding to one time slot, the following relation insures that the input bit rate can be accommodated within the time slot time less the guard time,

$$R\ t_D^{MAX} \geq B\ D = R\ t_D \tag{9}$$

where R is the bandwidth on a link, B is the bit rate for one circuit, t_D is the time interval for information transmission. and D is the duration of a frame from Eq.(8). The left side of Eq.(9) equals the maximum available number of bits for transmission during one time slot, while the right side represents the number of bits accepted for one circuit from the data terminal. Equality occurs when the whole t_D^{MAX} is used for transmission.

By simple manipulation of Eqs. (1,9),

$$t_{slot} = \frac{t_{idle} + \Delta t_D}{1 - \frac{KB}{R}} \tag{10}$$

where $\Delta t_D = t_D^{MAX} - t_D$ and adding Eq. (8),

$$D = K \frac{t_{idle} + \Delta t_D}{1 - \frac{KB}{R}}. \tag{11}$$

Since larger K results in a larger number of circuits available at a given blocking probability (Eq. (7)), we see the promised tradeoff between throughput and delay, where as expected the delay increases as throughput increases. In addition, KB is the total available throughput passing through any cross-point and is limited by the link bandwidth R; as a result, K can not be arbitrary large and is bounded by $\frac{R}{B}$. Also, Δt_D can not be reduced to 0 in practice, since some preamble bits are necessary for the bit timing recovery at the receiver. But it is much smaller than t_{idle} and can be neglected.

Figure 14. Compression delay as a function of the total number of time slots in a frame, where each slot is a 64 Kb/s channel, the link rate is 100 Mb/s, and the blocking probability is 0.001.

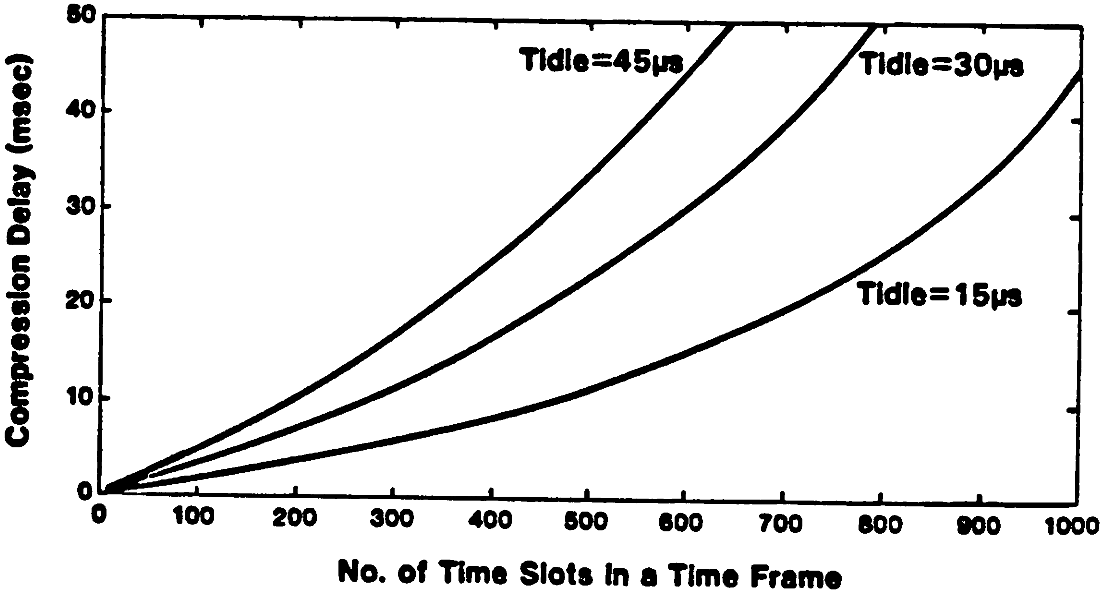
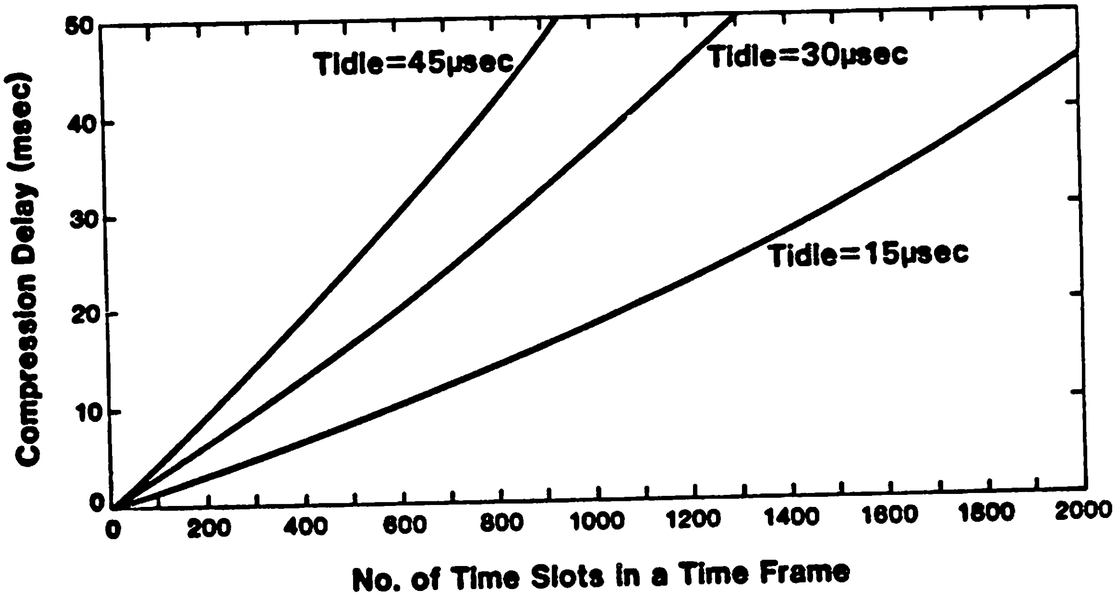


Figure 15. Same analysis as in Figure 14, but each slot is 16 Kb/s Channel.



Figures (14,15) give numerical results describing these relations for B = 64 Kb/s and 16 Kb/s respectively. We expect that R = 200 Mb/s can be achieved using a low-cost CMOS technology for the switches [2,5], but to be conservative we use R = 100 Mb/s. In the figures, Δt_D is assumed 0 in Eq. (11). If a propagation velocity of $2\cdot 10^8 m/sec$ is assumed in the optical fiber and the worst case in Eq. (5) is assumed, then the t_{idle} to network size ratio is 15 μsec/Km in figures (14,15). The three cases shown correspond, therefore, to a maximum network dimension of 1,2, and 3 km.

2.4.4. Network Utilization vs. Compression Delay

The previous results can be used to obtain the relationship between network utilization and compression delay if we combine the results for compression delay and blocking probability. In the following, assume that the full t_B^{MAX} in each time slot is used in transmission. For this case, by Eqs. (1), (2), and (10), we have:

$$\eta_{slot} = \frac{KB}{R} \qquad (12)$$

Furthermore, total network utilization can be easily expressed in terms of A, the average offered traffic in Erlangs. Because there are A circuits on average, by definition of network utilization, we have:

$$\begin{aligned}\eta_{network} &= \frac{AB}{R} \cdot \\ &= \frac{A}{K} \cdot \eta_{slot}\end{aligned} \tag{13}$$

By Eq. (7), we have:

$$\eta_{network} \approx N_2 \left\{ 1 - (1 - P_{bk}^{\frac{1}{K}})^{1/2} \right\} \eta_{slot} \tag{14}$$

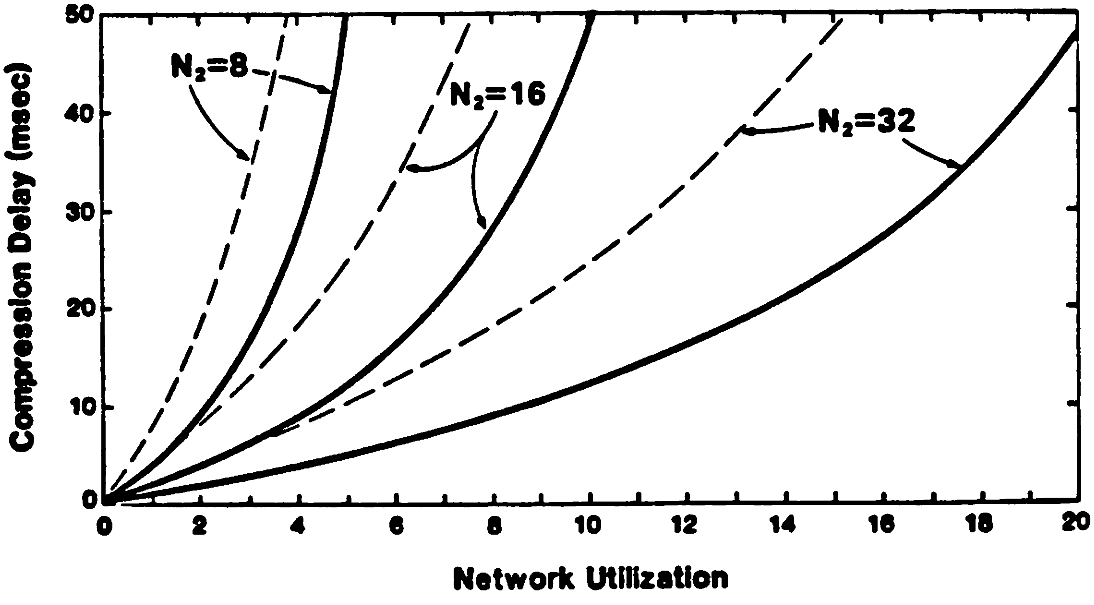
Eq. (14) gives very important physical insight into TSS networks. Total network utilization is naturally increased by the factor N_2 due to the muliple transmission links, while it is subject to two degrading factors. One is the effect of the guard time expressed by η_{slot}, and the other is the blocking probability limitation in the brackets.

By combining Eq.(11) with Eqs. (12,14), an expression relating D and $\eta_{network}$ can be obtained. Instead of writing this complicated expression, by observing:

$$\alpha \equiv 1 - (1 - P_{bk}^{\frac{1}{K}})^{1/2}$$

is a weak function of K, simple manipulation of Eqs. (7,11,13) gives

Figure 16. Trade-offs between compression delay and the network utilization, where each slot is 64 Kb/s, link rate is 100 Mb/s, and blocking probability is 0.001. Solid lines are for a network of 1 Km (15 μsec idle time), and dash lines correspond to 2 Km.

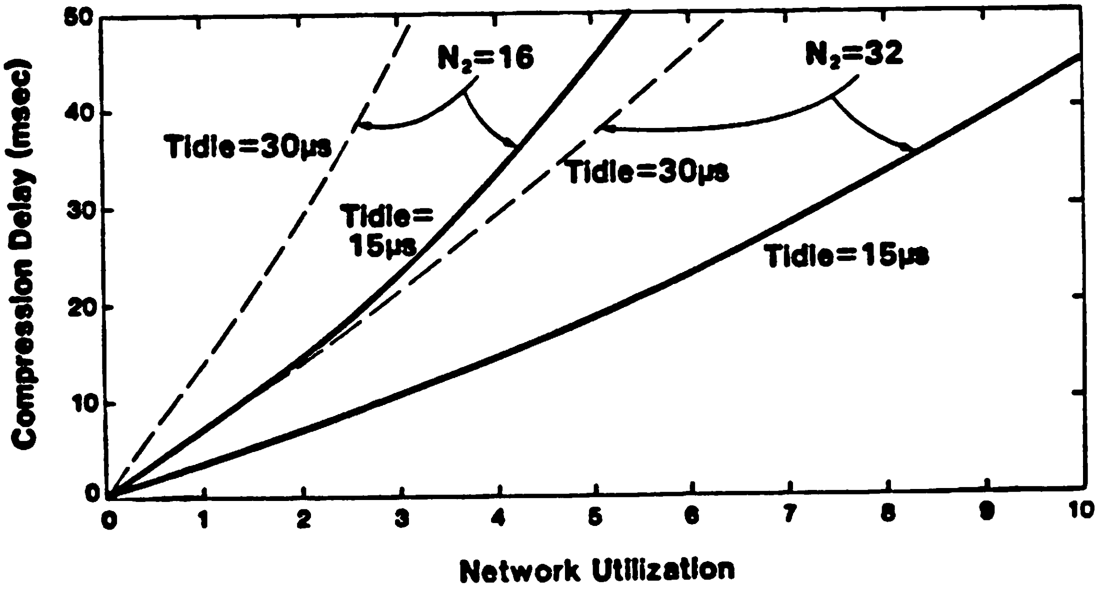


$$D = \frac{Rt_{idle}}{B}\frac{\frac{\eta_{network}}{N_2\alpha}}{1-\frac{\eta_{network}}{N_2\alpha}} \tag{15}$$

This expression is very similar to those in multiple access protocols, even though the underlying reasons are quite different. Figures (16,17) show that $\eta_{network}>1$ can be achieved at a reasonable delay; that is, the total utilization can be much larger than the bandwidth of one link that would be characteristic of a multi-access protocol. For example, at a 50 msec compression delay (which would be acceptable for a voice network), 100Mb/s network bandwidth, 10^{-3} blocking probability, and a 16 by 16 switch in the central star of a double star network, $\eta_{network}=10$ is easily achieved for 64 Kbps/slot. This corresponds to a 1000 Mbs total network throughput in comparison to a 100 Mbs for each link. Thus, for these parameters the increased capacity from multiple links dominates over the diminished capacity due to guard times and blocking probability.

The effect of the link bandwidth on the delay-utilization characteristic can be also observed from Eq. (15). With a given network utilization and fixed idle time, the delay is proportional to the link bandwidth. Intuitively, to keep the same utilization, the time interval in each time slot for transmission must be same,

Figure 17. Same analysis as in Figure 16, but 16 Kb/s channel per slot is assumed.



and the total bits transmitted in each slot is proportional to the link bandwidth, but the compression delay will also linearly increase. From Eq. (15), the effect of increasing link bandwidth is the same as decreasing the idle time by the same proportion. As a result, smaller idle time (smaller network) can be used to counterbalance the increase in delay because of increase in link bandwidth.

2.5. Heterogeneous TSS

In section 4, the homogeneous case where all circuits had the same bandwidth was considered. However, in practice we seek to integrate data, voice and video traffic within TSS.

First, we can have the following assignment:

1. Each time slot is equivalent to a 16 Kb/s channel.
2. Voice traffic uses a fixed 64 Kb/s bit rate, and each channel is assigned 4 time slots.
3. Interactive data traffic has a fixed rate, e.g. 16 Kb/s, 32 Kb/s, etc.; a multiple number of slots are assigned depending on its bit rate; that is, one slot for 16 Kb/s, and so on.
4. Video has no standard rate and is in the order of 20 Mb/s to 200 Mb/s. For simplicity, a whole frame circuit will be assigned for this large bandwidth traffic. Because the circuit switches are distributed and provide multiple circuits, this assignment will not significantly affect other traffic transmission.

5 Fixed length data such as file transfer can be assigned a certain number of slots based on its length. This will be explained in more detail in the remaining section.

Several characteristics of this approach should be mentioned:

1. If it can be arranged for these slots to be contiguous, then there is a potential saving in guard time. In the extreme case of a high bandwidth full-motion video signal, one entire link or most of a link could be dedicated to one circuit. The overall utilization would then increase considerably above that estimated in the previous section.
2. For the heterogeneous case, the compression delay remains equal to one frame interval.

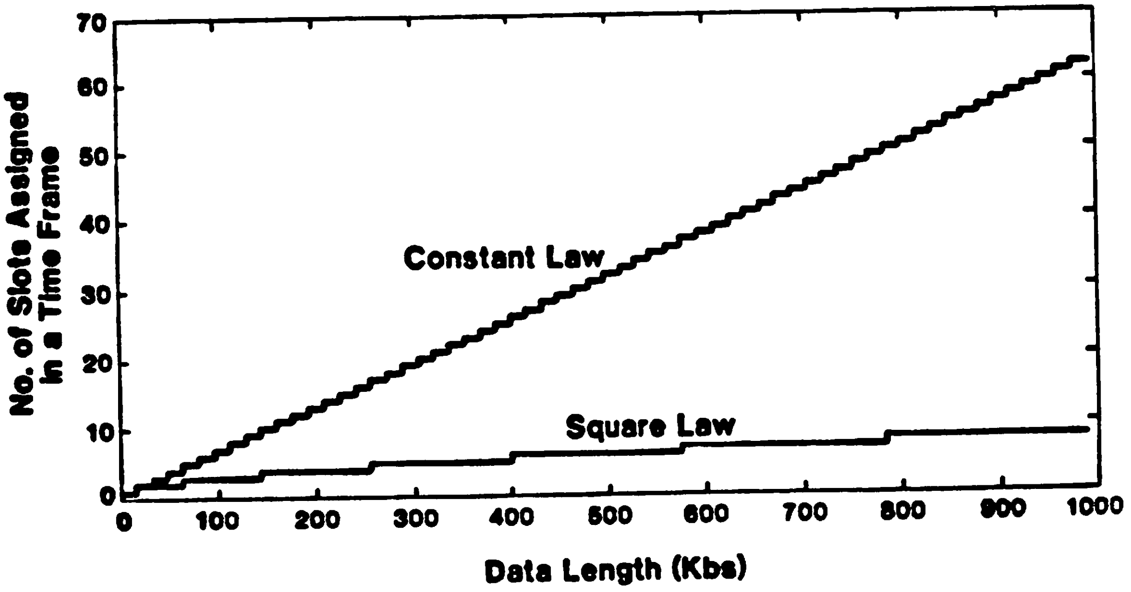
3. When the blocking probability for one circuit is small, say 10^{-3}, the blocking probability for a higher rate circuit made up of lower rate circuits is approximately multiplied by the number of time slots assigned in a frame.

4. For file transfer, we simply want the maximum bandwidth available to minimize the time to transfer the file. If there is only one file transfer request, and the file can be transmitted within one frame by available time slots, the number of time slots assigned can be variable depending on the size of the file, and the blocking probability is not a meaningful concept. However, if there is more than one request for file transfer, or the file transfer cannot be finished in one frame, the maximum use of available bandwidth will possibly block others' requests. Thus, we should be conservative in assigning bandwidth for file transfer.

For the last point relating to file transfer, we can suggest two approaches to assigning bandwidth. The first rule, the constant law,

$$n = \left\lceil \frac{L_D}{t_0 16Kb/s} \right\rceil \tag{16}$$

Figure 18. Number of time slots assigned as a function of the file length for file transfer.



assigns a bandwidth proportional to the file length (resulting in a constant file transfer time). The second, the square root law, assigns a smaller bandwidth,

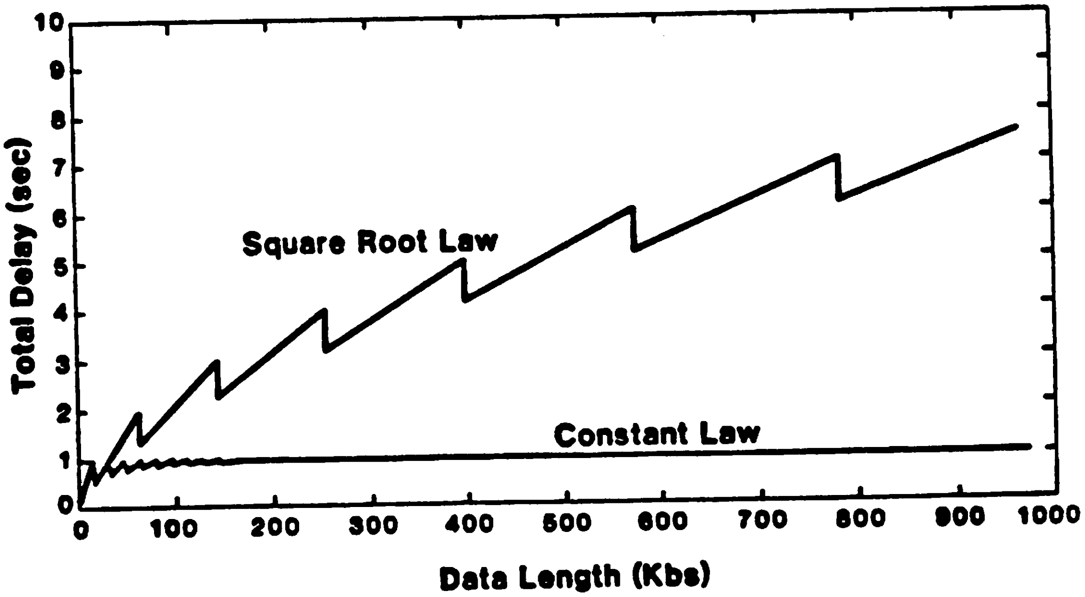
$$n = \lceil (\frac{L_D}{L_0})^{1/2} \frac{L_0}{t_0 16Kb/s} \rceil, \qquad (17)$$

where n is the number of time slots assigned which will be the same for each succeeding frames until the file is exhausted, L_D is the data size of the file, t_0 is a time parameter that is equal to the total transmission delay for the constant law, and L_0 is a length parameter. Transmission delay is equal to mt_0 when $L_D = m^2 L_0$ at square root law, and m is an integer. The resulting time for file transmission is,

$$\text{Transmission Delay} = \begin{cases} t_0, \text{ constant law} \\ (\frac{L_D}{L_0})^{1/2} t_0, \text{ square root law} \end{cases} \qquad (18)$$

Numerical results are shown in figures (18,19). The constant law promises smaller transmission delay (Eq. 18) at the expense of larger blocking probability for other requests (by assigning more free time slots). The square root law has less effect on other users.

Figure 19. Total time for file transmission as a function of the file length. Square root law has longer transmission time than constant law, but with a lower blocking probability.



The performance and hardware cost are attractive for voice and video traffic in this heterogeneous TSS since they are well suited to circuit switching. The same can be said for file transfer activity as long as the file length is long enough that the circuit connection time is insignificant. For interactive data traffic, however, there are significant disadvantages:

1. For very small data packets, the overhead in establishing a circuit is too large.
2. Interactive data traffic at a fixed rate is idle most of the time. The slots reserved are wasted during idle periods. This reduces the effective network utilization below our earlier estimates.

If these disadvantages are dominant, as in a network where interactive data represents a significant fraction of the total offered traffic, it is possible to overlay packet data networks on top of TSS. For example, we can establish a packet network operating at lower speeds where the implementation costs are reasonable by establishing semi-permanent circuit links through the TSS network. For example, a token ring can easily be established. TSS is very flexible in its ability to reconfigure and reassign capacity to various services such as packet data networks on a demand basis. It is also possible to combine TSS with a CSMA/CD data network in a portion of the frame [6,7]. These extensions of TSS to combine CSMA/CD or slotted-ring will be considered in Chapters 3 and 4. In fact, the presence of the active switches considerably simplifies the detection of collisions in such a network. This extension of TSS will be explained in chapter 3. We should emphasize again that due to the asynchronous nature of the switches, overlaid data networks can also use arbitrary bit rates as long as they adhere to the maximum rate imposed by the switch.

2.6. Conclusion

In this chapter, we describe a new protocol for local-area networks: time slot switching or TSS. Since electronics is the factor limiting the bandwidth of practical fiber optical networks, TSS is very attractive by minimizing electronics in switching nodes. It uses circuit switching, which appears more appropriate for very high speeds and is quite compatible with voice and video traffic, but does not preclude overlaid packet networks for interactive data traffic operating at more moderate speeds. The network is compatible with fiber optics technology, since it utilizes only point-to-point links at speeds that are quite modest by fiber standards. It is quite compatible with integrated traffic, and can easily provide different effective speeds for different services. Its greatest strength is the ability to incrementally add capacity and the ability to reach quite impressive total throughputs without exotic technologies. Its greatest weakness is its limitation in geographical size to the order of one to two kilometers. This latter limitation can be overcome by the standard technique of adding gateways with buffering at the expense of additional delay.

TSS can be described as a wideband distributed PBX, and is perhaps a closer relative to today's PBX products than it is to traditional LAN approaches. Since technologies springing from both LAN and PBX products show promise in providing an integrated solution to local communications, it will be interesting to see which approach becomes dominant.

Appendix A. Blocking Probability Analysis: Single Star

In this appendix, the blocking probability is estimated based on Lee's method [3] for a single star network as shown in figure (6).

Assume that p, the probability of any one time slot on any one link being already used, is known, and that the events of different time slots on the same link or different links being used are independent. Now, suppose we want to form a new connection between two TCMs in any particular time slot. The probability of success is $(1-p)^2$. Therefore the blocking probability for the connection in any particular time slots is $1-(1-p)^2$. Because there are K time slots can be chosen for the connection, the probability that they are all blocked is

$$P_{bk} = \left\{1-(1-p)^2\right\}^K \tag{A.1}$$

It remains to determine the probability p. Assume there are A time slots in use on average, then the active probability for user terminals is $p_0 = A/NM$, since there are total NM users. Assuming the traffic is uniformly distributed in all time slots and there are N user terminals in each TCM,

$$p = p_0\frac{N}{K} = \frac{A}{MK} \tag{A.2}$$

Appendix B. Blocking Analysis: Double Star

The blocking probability for the double star in figure (10) is estimated in this appendix. It is assumed here that a circuit connection follows the same hierarchy principle as in telephone networks. That is, if a circuit can be locally connected through the local switching star, no connection will be required in the central switching star. Therefore, a blocking probability is larger if a circuit needs to go through the central switching star. To obtain a conservative estimate, this latter probability will be used.

By Lee's method and using the same technique as in Appendix A, the blocking probability is

$$P_{bk} = \left\{1-(1-p_1)^2(1-p_2)^2\right\}^K \tag{B.1}$$

where

$$p_1 = \frac{A}{K(N_1+1)N_2}$$

is the probability of use for the time slots around local switches, and

$$p_2 = \frac{A}{KN_2}$$

is that probability for the time slots around central switches.

Since p_1 is in general much smaller than p_2, the factor $(1-p_1)^2$ in Eq. (B.1) can be neglected. Network throughput may therefore be expressed as:

$$A \approx KN_2 \left\{ 1-(1-P_{bk}^{\frac{1}{K}})^{1/2} \right\} \tag{B.2}$$

REFERENCE:

[1] Kamilo Feher, *Digital Communications, satellite/earth station engineering*, chap. 8, written by S. J. Campanella and Daniel Schaefer, Prentice-Hall press, 1981.

[2] H. J. Shin and D. A. Hodges, *CMOS Switch for 200 Mb/s Fiber Optic Networks*, to be published.

[3] C. Y. Lee, *Analysis of Switching Networks*, BSTJ, Nov. 1955, pp1287-1315.

[4]Charles Clos, *A Study of Non-Blocking Switching Networks*, BSTJ March 1953, pp. 406-427.

[5] G. A. Hayward, A. M. Gottlieb, D. G. Boyer, and J. E. Berthold, *High Speed 16x16 CMOS Crosspoint Switch*, Electronics Letters, Vol. 21, No. 20, 1985, pp. 923-925.

[6] Ming-Kang Liu, David G. Messerschmitt, *An Integrated Network from TSS and CSMA/CD for Data/Voice/Video LAN*, to be published.

[7] Ming-Kang Liu, David G. Messerschmitt, and David A. Hodges, *An approach to Fiber Optics DATA/VOICE/VIDEO LAN*, INFOCOM 86, pp. 516-523.

CHAPTER 3

Integration of TSS and CSMA/CD

3.1. Introduction

In chapter 2, the concept of time slot switching (TSS) is proposed to allow all data, voice, and video traffic in a single optic network. As mentioned, TSS is ideal for voice and video traffic because of its circuit switching nature. However, as pointed out in chapter 1, data traffic is very bursty and has low activity; as a result, most data networks are packet switched instead of circuit switched. CSMA/CD, ALOHA, and Token-passing are examples [18-20]. Based on above considerations, an integration of TSS and CSMA/CD is motivated.

To justify the intergration, in section 2, a comparison between the performance of TSS and CSMA/CD for the interactive data traffic is first investigated. The results prove that CSMA is better than TSS for this traffic type in terms of delay vs. available circuits. Therefore, TSS combined with CSMA is promising and its operations are explained in section 3. The basic hardware architecture and part of the operational principles are similar to the original TSS network except that CSMA/CD is alternatively operated with TSS in the same network. During the CSMA cycle, the circuit switches will be operated in a broadcasting mode. In section 4, some features of integrated TSS/CSMA are discussed. First, by the special property of the switching matrix, collision detection circuitry can be replaced by energy detectors. Second, the performance and the flexibility in changing the cycle ratio are also addressed. All these show that integrated TSS/CSMA is satisfactory for mixed data, voice, and video environment.

3.2. A Performance Comparison between TSS and CSMA

As a comparison of the performance between TSS and CSMA, interactive date traffic (e.g. terminals and hosts' conversation) is studied in this section. The figure of merit is the number of interactive links which can be provided for the same link bandwidth and the same transmission delay. The quantitative result justifies integrated TSS/CSMA in a single network.

There are many analyses of the performance of CSMA [1,2], assuming an infinite number of users with finite total incoming rate obeying Poisson process. They also have non-persistent or p-persistent re-access versions when collision is detected [2]. There are other analyses of modified CSMA/CD disciplines, for example with virtual time [3,4] or with priority queue [5,6].

For simplicity, we will use the analytic results from [1] for the comparison. This is a more conservative comparison than in [3-6]. (If the results favor CSMA, of course this implies modified CSMA in [3-6] is also favored, since it is better.) In the analysis of [1], both a finite total incoming Poisson rate and a fixed successful access probability (the probability that there is no collision) are assumed. This requires some traffic monitoring and feedback mechanisms to adapt the access rate of each station wishing to access the channel. For example, a binary exponential backoff algorithm can be used [7].

The comparison between CSMA and TSS using results from [1] is shown in figure (1). Both network sizes of 10 Km and 1 Km are compared. (We assumed 5 μsec per Km of fiber propagation delay.) In the

Figure 1. Performance Comparison of TSS and CSMA by [1]. Two network sizes are 1 Km and 10 Km, which correspond to 5 μsec and 50 μsec. (see text). TSS results are under the same conditions of Figure 2.

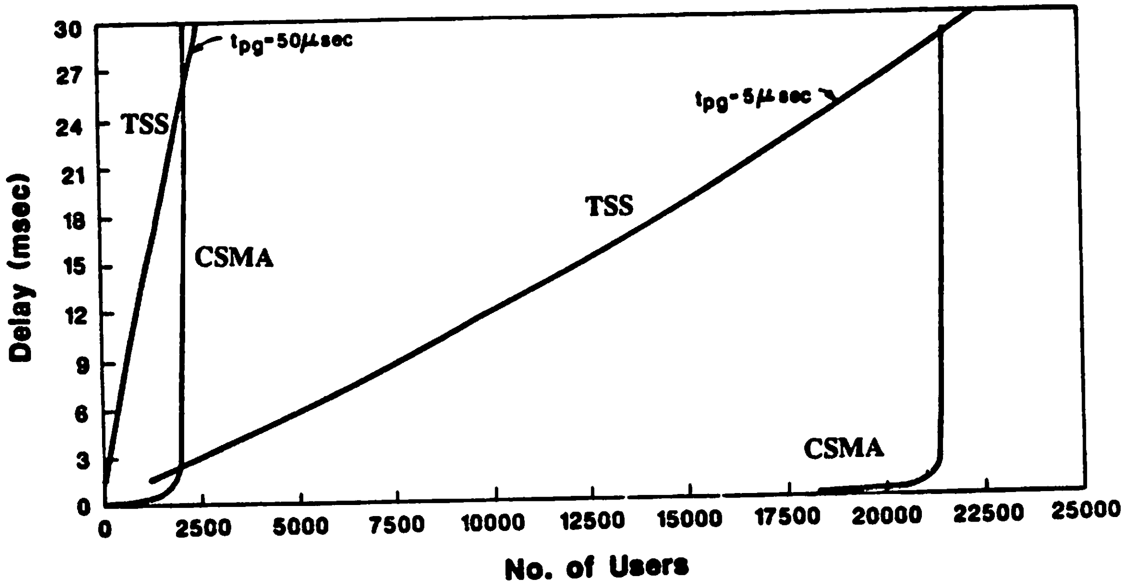


figure we see an increase in delay when traffic intensity increases (which is proportional to the total number of conversations. We assumed a mean Poisson rate for each user of 1 packet/sec. The total number of conversations is therefore equal to the ratio of the total Poisson rate to the individual Poisson rate.) Apparently, the result shows CSMA is better than TSS until the saturation point is reached.

In this chapter, another CSMA model for interactive data traffic is assumed. This is to provide a better practical model than [1] for interactive data traffic. If someone is typing a character or sending a command, he will not generally type another character or send another command. Also, a traffic adaptive mechanism is not always available. Even if there is, it might not be ideal to provide a fixed successful probability. The detailed model is studied in the Appendix. Briefly, it is assumed there are finite users sharing the common CSMA network. Each has the same Poisson access rate. But whenever a user has a packet to send, but not yet finished, there is no other new incoming packet.

Figures (2,3) show the results from the simulation of this model. Figure (2) shows a comparison among the results from TSS, CSMA by [1], and the simulation, where the successful access probability is also assumed fixed in simulation. It is noted that CSMA by [1] and the model in simulation have approximately the same saturation point. However, for the model in simulation, because of smaller total Poisson access rate when longer queuing, the slope is much smaller than [1]. Because of the bursty nature, both cases of CSMA are better than TSS before saturation.

Figure (3) gives the simulation results when the access rate for each individual user is fixed. (Therefore the successful access probability decreases when total traffic increases.) Different curves obtained from simulation correspond to different access rates for each individual. It seems there is an optimum access rate, as explained in the Appendix. From the figure, note that CSMA is still more attractive than TSS if a good access rate is assumed.

The above results suggest that CSMA is more ideal for data traffic than TSS. On the other hand, if conversation links are used with highly active percentage, circuit switching by TSS is much better, as explained in chapter 2. This motivates the integration of TSS with CSMA. The remaining sections of the chapter are devoted to this integration.

Figure 2a. Performance comparison of TSS and CSMA by the simulation. The model is described in the appendix, where fixed successful access probability is assumed. The curve by [1] is also drawn for comparison. The network size is 1 Km. The TSS result is under blocking probability of 0.001, double star topology, 16 by 16 switches, and 16 Kb/s per slot.

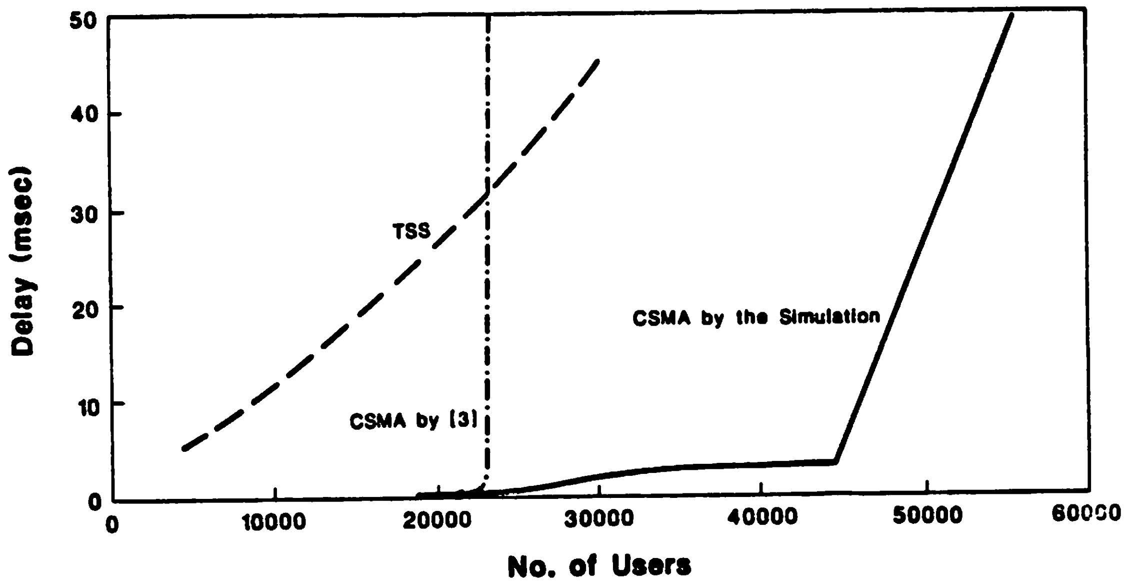


Figure 2b. Same as Figure 2a, except that the network size is 10 Km.

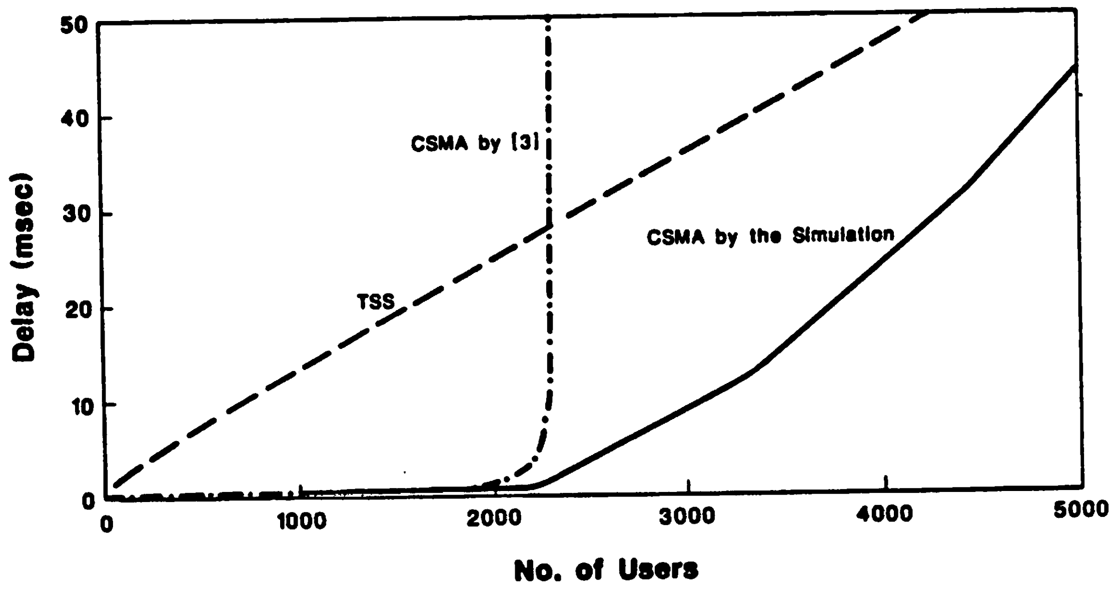


Figure 3a. Performance comparison of TSS and CSMA by the simulation, where fixed access rate for each individual user is assumed. The network size is 1 Km. Other assumptions are the same as Figure 2.

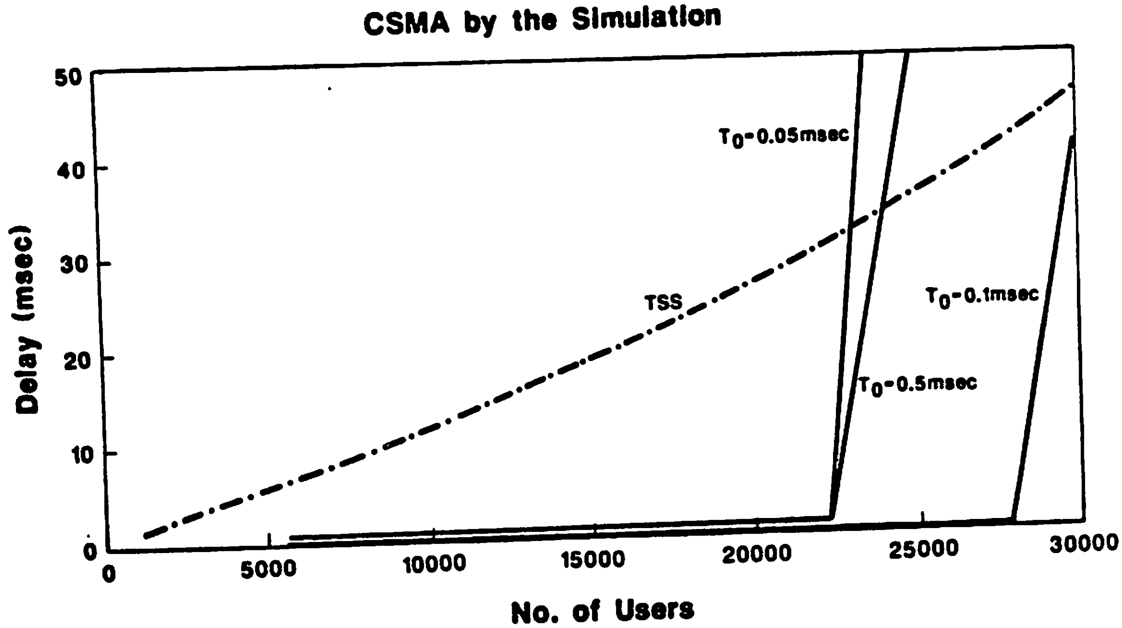
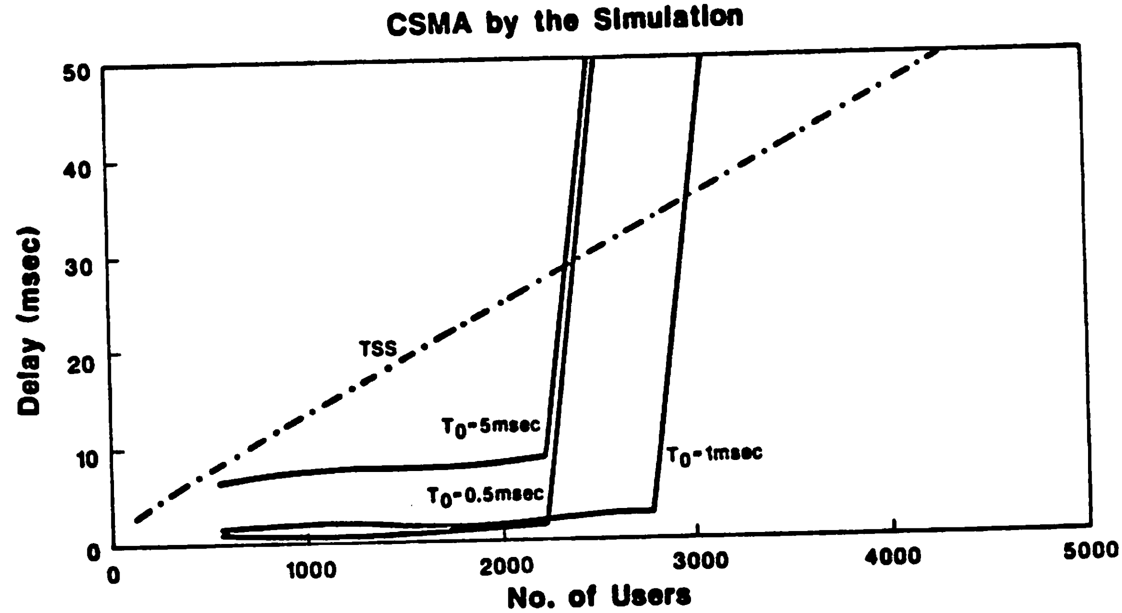


Figure 3b. Same as Figure 2b, except that the network size is 10 Km.



3.3. Medium Access Protocol for Integration of TSS and CSMA

In this section, we discuss how to integrate CSMA/CD and TSS in a single network. Basically, the network is still in a tree or double star topology suggested in chapter 2. In the original CSMA/CD network, it is a bus topology where the packets are broadcasted. By having the capability of broadcasting from circuit switching in a TSS network, CSMA/CD can be easily added. A simple medium access protocol is suggested for the integration:

(1). Periodical Time Frames:

Each time frame is divided into two subframes: one for TSS switching, the other for CSMA broadcasting, figure (4). Let us call the first part **"Switching Cycle"** , and **"Broadcasting Cycle"** for the second part. The same structure of time frames is repeated.

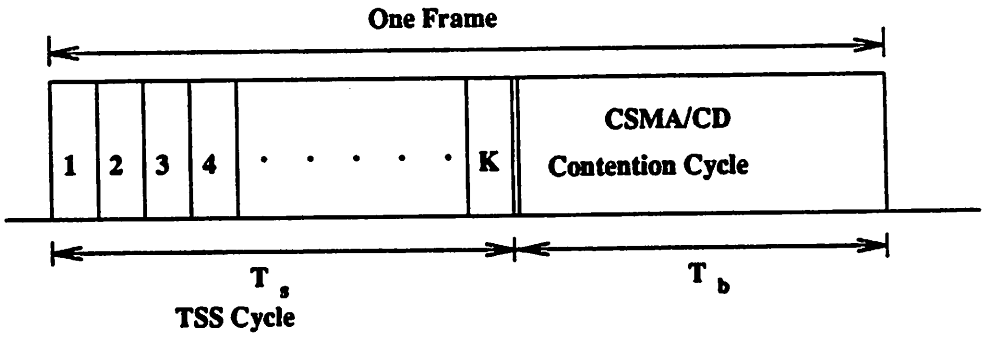
(2). Switching Cycle for TSS:

In each switching cycle, we still have many time slots. These time slots are assigned for voice traffic and certain types of data. Data of medium size (1Kbs to 100Kbs, for example) is in this category. Large size data whose transmission delay is not very important can also be assigned in this category. Time slot assignment is the same as suggested in chapter 2.

(3). Broadcasting Cycle for CSMA/CD:

In each broadcasting cycle, the switching matrix in each concentrator is connected in a broadcasting configuration. This will be discussed in detail in the next subsection. Users desiring to transmit certain types of data contend in this cycle. Small size data packets (below 1Kbs) and large size packets (above 100Kbs) requiring short transmission delay are generally in this category. We assign small

Figure 4. Time domain picture of integration of TSS and CSMA, where each frame consists of two subframes: one for TSS and one for CSMA.



packets because of its inefficiency in connecting a call by the central control for a very short duration, and large size packets because of very long transmission delay by TSS.

(4). Cycle Time Adaptation:

The ratio of switching cycle and broadcasting cycle (ratio of T_s and T_b in figure (4)) can be either static or dynamically changed. In this way, the network can always be tuned to its optimum performance, based on the specific traffic distribution of the network.

(5). Transition Processing between Two Subframes:

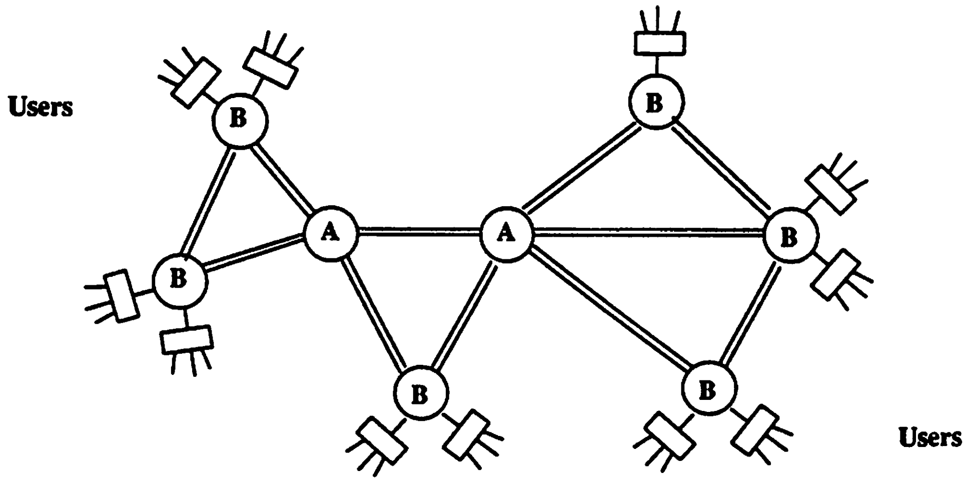
If packet transmission is not completed at the end of CSMA broadcasting cycle, the packet will be repeated in the coming broadcasting cycles.

By integrating these two different services in a single network, some interesting new features are discussed in the following section.

3.4. Further Considerations

When combining the two protocols, TSS and CSMA, as suggested in the last sections, some new aspects arise which are different from considering each protocol alone. These include the consideration of

Figure 5a. Network topology for the integration of TSS and CSMA



hardware implementations, flexibility in adapting the cycle ratio of TSS and CSMA, and also the individual performance after the integration.

3.4.1. Hardware Architecture

The overall network architecture can be shown in figure (5). Figure (5a) shows a star-like topology, which is just the same as that of pure time slot switching networks (in chapter 2). For the switching nodes of type A, where there are no users connected, the specific architecture is shown in figure (5b). As compared to TSS networks, the control network for TSS signaling can now be handled by the CSMA subnetwork, instead of some dedicated slots as suggested in chapter 2. For the switching nodes of type B, where there are users connected, the specific architecture is shown in figure (5c). This shows it is very similar to the architecture in pure TSS. The only difference is the extra CSMA subnetwork link level processing.

The circuit switches in the node physical level provide the data path for both TSS and CSMA. The switching matrix in each switching node can not only be operated for circuit switching, but also can be

Figure 5b. Architecture inside Class A node.

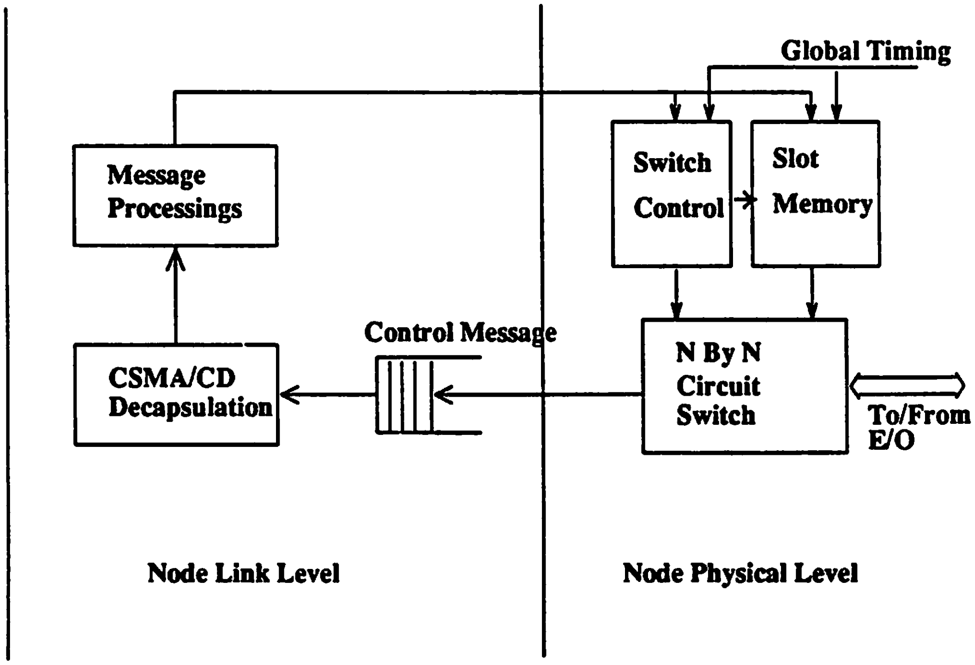


Figure 5c. Architecture inside Class B node.

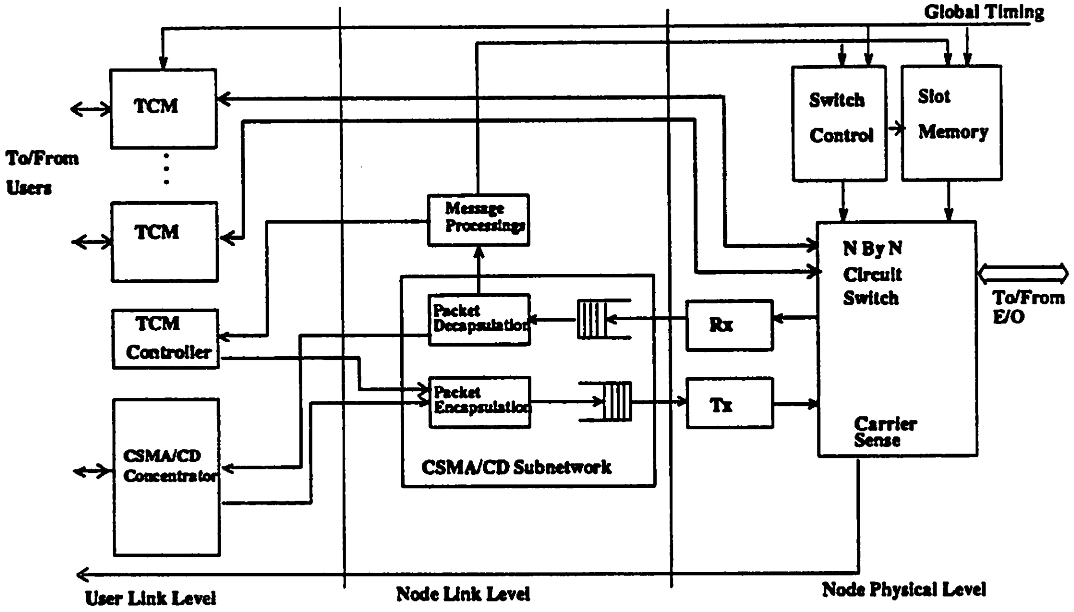
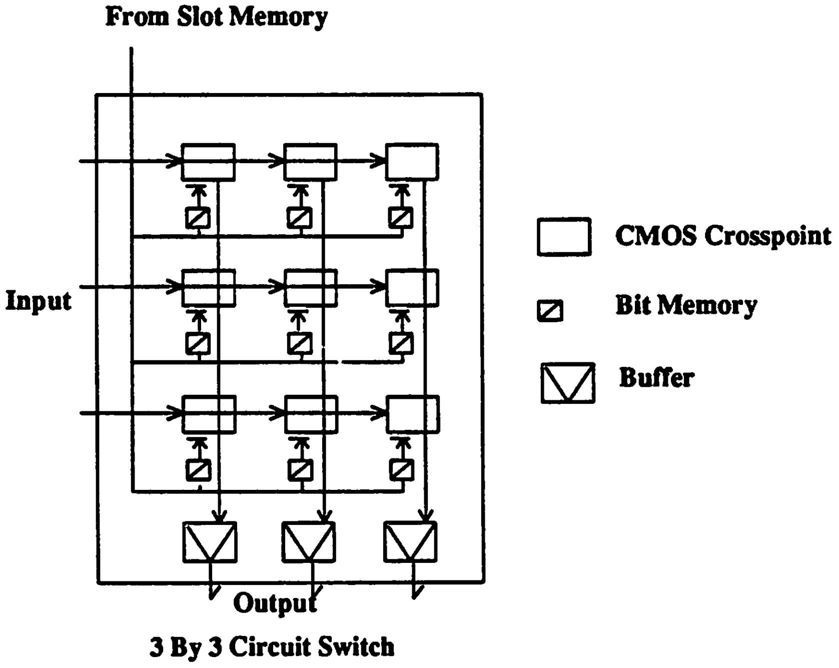


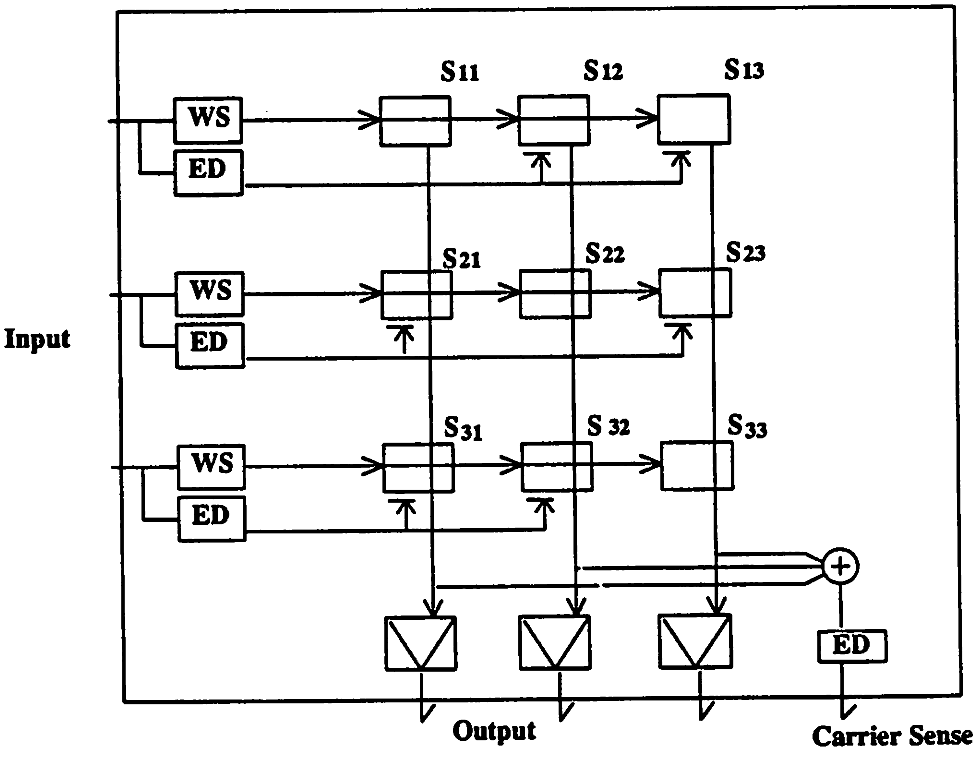
Figure 6a. Architecture inside the switch where TSS is under operation. The switches are controlled by the bit and slot memory, which is written by the data from the central controller.



configured for broadcasting. Figure (6) shows the switch architecture illustrated by a 3 by 3 matrix. Figure (6a) shows the configuration of the switching mode. Its operation has been described in chapter 2. During the broadcasting cycle, each switch is controlled by an energy detection circuit, as shown in figure (6b). When there is an incoming signal to some channel, say channel 1, the switches S_{12} and S_{13} are connected by the signal from energy detector ED_1. A broadcasting configuration is thus formed.

Generally speaking, multiple switching nodes in a TSS network can provide multiple paths for a broadcast packet, figure (7a). Because of the delay difference between different paths, "self collision" can happen if care is not taken. This problem can be easily overcome by linearizing the network; that is, cer-

Figure 6b. Architecture inside the switch where CSMA is under operation. The switches are controlled by the energy detector, which will close the switch if there is an incoming signal.



3 By 3 Circuit Switch

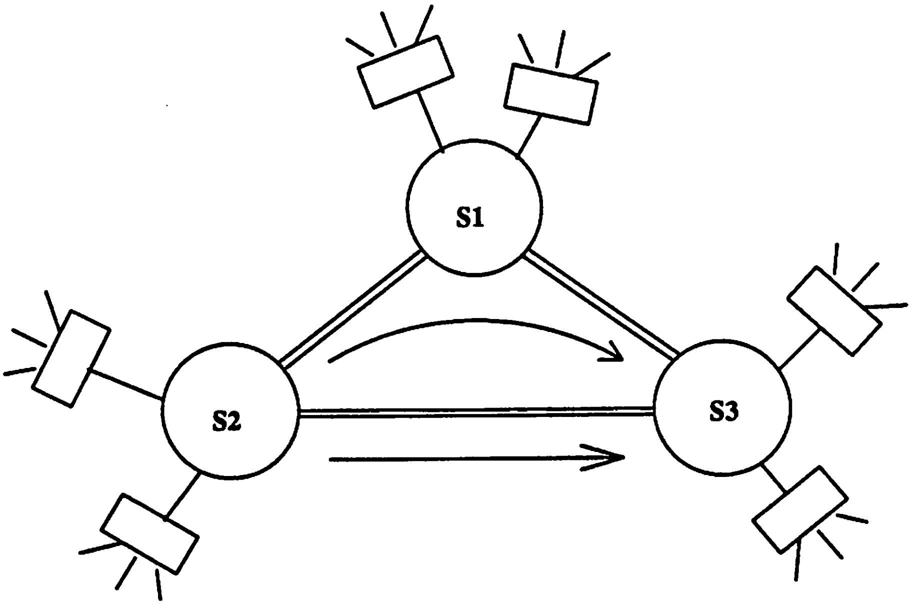
WS: Waveform Shaping ED: Energy Detection

tain crosspoints in the circuit switches are always disabled so that packets can be broadcast in only one direction, figure (7b).

Because of the nature of CSMA/CD, collisions are possible. To recognize the collision and abort the transmission immediately, collision detection is required. There are different approaches to implementing this detection function [9-11]. Most of the methods suffer a high cost receiver to detect signals of large dynamic range or a high speed processor to decode the data.

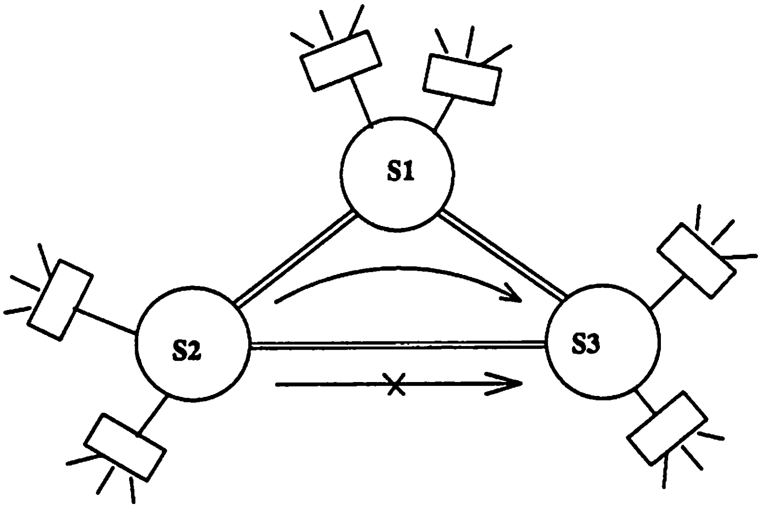
By using the circuit configuration inside the switch as in figure (6b), a simple collision detection mechanism can be implemented. Along the diagonal, each switch is always off in the broadcasting cycle. This prevents the transmitter from receiving its own packet. Whenever the corresponding receiver receives anything, a collision can be immediately recognized. This arrangement in fact has an optical analog in [12], but at the expense of fancy optical technology. This simplification of circuits for collision detection is consistent with the low cost motivation of TSS.

Figure 7a. Illustration of multiple paths for packet broadcasting.



Multiple Paths for Packet Broadcasting

Figure 7b. Linearization of a network to avoid multiple paths.



Crosspoints for connections between S2 and S3 are disabled in CSMA cycle to linearize the network

3.4.2. Traffic Priority

By integrating the different service characteristics of CSMA/CD and TSS in a single network, and including the flexibility of changing the ratio of T_s to T_b, the specific needs for each network can be satisfied. The ratio above can be either static or dynamically adapted, depending on the particular network environment. For example, in an office scenario, the demands for data and voice will most likely follow the same trend (both high in the peak hours, and both low near the beginning and end of each working day). A static ratio will be fine for this situation. On the other hand, for a research and development environment, the traffic requirement distribution may fluctuate widely. It is better if a dynamic division of the two services can be provided.

Some work on the dynamic traffic service division has been done, for example a Moving Slot Time Division Multiplexing (MSTDM) is proposed in [13]. Basically, the network follows the CSMA/CD protocol. By insisting transmission even with collision for voice traffic, this protocol behaves like TDMA. For data traffic, the protocol behaves just as CSMA/CD. This protocol has a good property of fixed delay for voice traffic, and ensures no packet will be lost. However, when voice traffic keeps increasing, the network capacity will be occupied by all the voice traffic. There will be no room for data traffic. Therefore this network can be described as a *"voice-priority"* network. In the following, a simple adaptation scheme is suggested in changing the ratio of two service cycles. By limiting the range of the ratio, the network can be operated either as a *"voice-priority"*, *"data-priority"*, or a *"balanced"* network.

Suppose there are total of K time slots in a time frame:

(1) Assume currently there are K_s time slots for the switching cycle, and K_b time slots for broadcasting cycle. Obviously, $K_s+K_b=K$.

(2) If there are N_{bk} incoming calls which are blocked, increase K_s by 1. K_b therefore is decreased by 1.

(3) K_s cannot be increased more when K_s reaches its upper bound $K_{s,max}$.

(4) If no calls are blocking during a time interval T_{free}, K_s is decreased by 1.

(5) K_s cannot be decreased more when reaches its lower bound $K_{s,min}$.

By defining the parameters $K_{s,max}$ and $K_{s,min}$, we can determine the priority for voice or data traffic. Also, by defining the parameters N_{bk} and T_{free}, we can determine the adaptation speed.

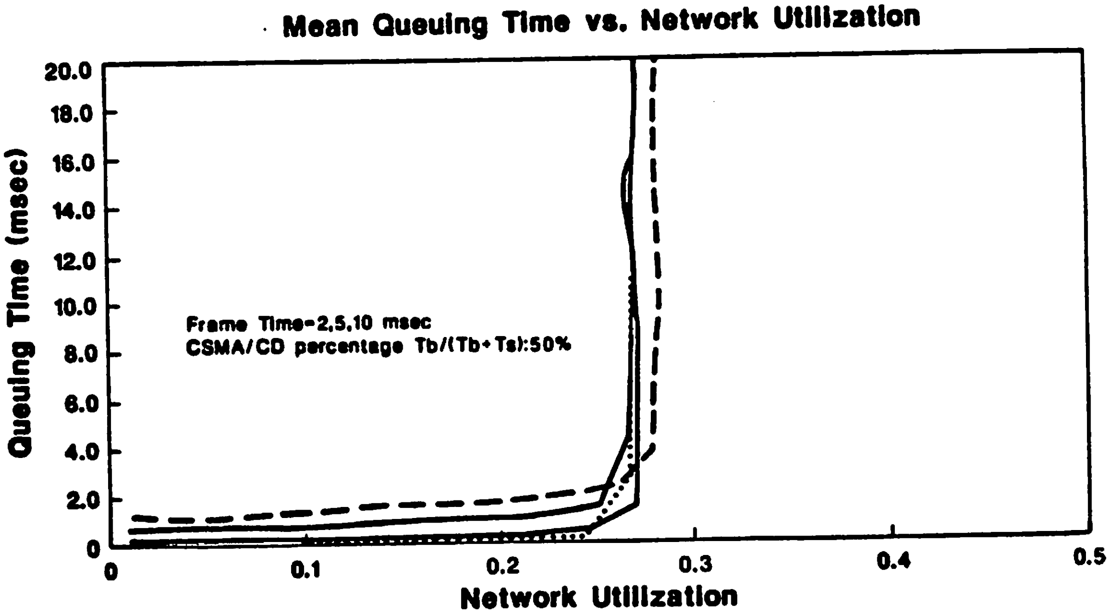
3.4.3. Traffic Performance

When TSS and CSMA/CD are integrated, each time frame is divided into two subframes for each access protocol, and the service from each protocol is no longer continuous. That is, TSS and CSMA/CD provides the service via the same communication media alternatively. This situation can be modeled as cyclic servers or vacation servers [14-17]. There are two classes in the cyclic server models: exhaustive and non-exhaustive. If the server keeps servicing until the queue is empty, it is called exhaustive. Otherwise it is called non-exhaustive. In the exhaustive case, there are also some subclasses. In the analysis developed in [17], the only difference for each subclass is the upper limit of the number which each server can serve in each cycle. For example, if the number is one for each server, it is called the "ordinary cyclic model". In all the above cases, the time interval or the cycle time for each server is stochastic. However, in the case of this chapter, the cycle time is fixed. (At least for the static adaptation case, and slowly varying for the dynamic case.) Careful examination of this problem suggests an analytic solution is not easily obtained. Instead, some simulation results and discussions are provided for the integrated performance.

Let us first concentrate on the performance of TSS. By assigning a fraction of each time frame for CSMA/CD, there are fewer time slots available than for single TSS service. If the amount of traffic demands from TSS is still the same, the blocking probability is of course larger. However, since some data traffic is now assigned to CSMA/CD, the demands from TSS should be less than for single TSS. If the data traffic is well divided between TSS and CSMA/CD, based on its characteristics, as compared in section 2 and suggested in section 3, the blocking probability might be even smaller. In any case, the transmission delay is the same as single TSS server, as long as a circuit link is found and assigned.

For the performance of CSMA/CD, the delay-utilization analysis is hard because the service time distribution is not a simple independent process. A data packet in transmission and near the end of a CSMA/CD cycle can be stopped because of insufficient time left. This probability depends on which point the customer began to be served in the CSMA/CD cycle and how large the data packet is. Although no

Figure 8. Performance comparison between pure CSMA and cyclic CSMA with period and active percentage. Except the insignificant extra delay before the saturation, they are almost the same. The results are obtained from the simulation.



easy model can be constructed for the analysis, some simulations results are shown in figure (8). In the simulation, fixed probability of successful access is assumed, as in the analysis of [1]. This can be realized for example by the binary backoff algorithm.

The results suggest the performance of integrated CSMA is almost the same as that of a single CSMA. The only difference is the extra delay before the network utilization is beyond the saturation point. By comparing the saturation points, they are almost the same for the integrated CSMA and single CSMA. By examining the extra delay before the saturation, it is quite close to the mean residual delay as in the exhaustive model [14]. This suggests the integration model in the chapter can be approximated to be an exhaustive cyclic server model. In the model, the mean delay is the mean delay for a single server, plus the mean residual delay. An intuitive argument for this is that the queue length at the end of each CSMA cycle is very small. This is because the service capability of CSMA is larger than the average one (when CSMA is continuously served).

From the above discussion, it is suggested that the integration of TSS and CSMA for mixed traffic environment is justified.

3.5. Conclusions

In this chapter a protocol integrating TSS and CSMA/CD is explained. The original motivation for the integration is to efficiently use the bandwidth for all data, voice and video traffic, by observing the significant difference in traffic characteristics. Since TSS is circuit switching and CSMA is packet broadcasting, it is natural to integrate these two in a single network. The integration is justified not only for performance, but also for implementation of high speed circuits.

From the performance point of view, there is the flexibility not only in assigning the traffic through either TSS or CSMA based on the traffic type, but also in changing the two service cycles ratio for the specific networks. This flexibility achieves the initial goal of mixing the two protocols. In addition, the individual performance for both TSS and CSMA is not quite different from each alone. For TSS, the transmission delay is the same when a circuit link and time slots are available. By removing some data traffic to CSMA, the blocking probability might be even smaller than before. For CSMA, there is a little extra tolerable delay from residual waiting time when the server is on vacation. The saturation point is almost the same.

From the implementation point of view, there is no significant extra cost of adding CSMA into the original TSS network. CSMA is actually put on top of the TSS network following the same network topology and using the same network communication links and components. The high speed circuit switch can also be used for broadcasting for CSMA. The cost in collision detection is almost avoided by using simple energy detectors. The only extra cost might come from additional processing for packets in CSMA and energy detectors in the switches. All these could be a small amount compared to the cost for the original TSS.

Although the integration of TSS and CSMA is very attractive, as observed in figures (1-3), the performance of both TSS and CSMA is very sensitive to the network size. To overcome this limitation to larger networks such as metropolitan area networks (MAN), a modified scheme of TSS called skew TSS (STSS) is proposed in chapter 4. In addition, to preserve packet switching in the same network as CSMA in TSS, an integration of STSS with slot-ring is also investigated in the next chapter.

Appendix. Simulation Model for Interactive Data Traffic by CSMA/CD

The following is the formulation for the simulation of CSMA, where a finite number of users is assumed. The purpose of this study is to give more practical results for interactive data traffic in CSMA. The simulation model, the algorithm, and the results are discussed in the following subsections.

A.1: Simulation Model

The simulation model is summarized as:

I. We have finite users with equal input Poisson rates of $\lambda = t_{arvl}^{-1}$.

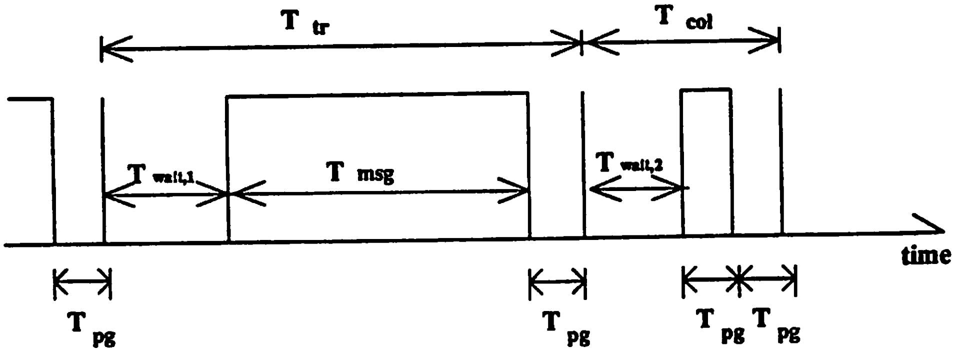
II. Modifications of access scheme, see figure (A.1) also:

(1) "Ready user" is defined for the user who has packets and desires access to the network.

(2) If the ready user senses an idle channel, it will start its transmission with a random time delay t_{wait} after the previous transmission or the detection of the idle channel. This time delay is uniformly and randomly generated from a time interval $[0, t_0]$. In the analysis, we assume all the users have this same distribution. While waiting, if another user starts transmission first, the ready user regenerates its waiting time and starts another waiting period.

(3) Because of the non-zero propagation delay t_{pg}, it takes t_{pg} to detect the end of the transmission of the current active node. This value is assumed to be the same for every user.

Figure A.1. Timing diagram for the CSMA simulated.



(4) A transmitting user will instantly abort its transmission when collision is detected. The retransmission is scheduled as in previous steps.

III. Service time distribution: It includes a random waiting time, packet transmission time, possible collision time, and constant propagation delay.

(1) Packet transmission time can be any distribution, but with a minimum packet size of 2 t_{pg} to guarantee the detection of possible collisions. In the simulation of this chapter, a constant size of 2 t_{pg} is assumed.

(2) The number of possible collisions before successful transmission is a geometric distribution, with the successful access probability as the parameter, which depends on the traffic intensity.

(3) The random waiting time is uniform, as assumed in (II). Each collision time is independent and follows the second order-statistics of the waiting time distribution. This can be understood as follows. The user with minimum waiting time wins the right for transmission. If the second minimum waiting time is greater than the first minimum waiting time by a propagation delay time t_{pg}, we shall have a successful transmission. Otherwise, a collision will be observed since it takes t_{pg} to detect the transmission of other nodes.

A.2: Simulation Algorithm

In this subsection, a time domain picture for the simulation is described. The relationship between the access rate and the successful access probability at a given traffic intensity is also derived.

(1) Assume there are a total N users are using the network for interactive data communication. There are only k users with packets to transmit, and the remaining users are currently idle.

(2) A timing picture for CSMA/CD is shown in figure (A.1), where T_{wait} is the waiting time after the detection of the end of the previous transmission. It is a random variable in $[0, T_0]$.

(3) T_{mg} is the random variable for message duration with a certain distribution. Therefore, by figure (A.1), we have:

$$T_{TR}=T_{wait,1st}+T_{mg}+T_{pg} \qquad (1)$$

(4) The condition for collision is: $T_{wait,1st}+T_{pg}>T_{wait,2nd}$. The reason is that if $T_{wait,ith}>T_{wait,1st}+T_{pg}$, for i=2,3,..., they will notice the transmission of the user with minimum T_{wait} before their transmissions. Consequently:

$$T_{COL}=T_{wait,2nd}+T_{pg}+T_{pg} \qquad (2)$$

We put T_{pg} in each of the above terms because this is the time needed before a new transmission period.

(5) In the simulation, $T_{wait,1st}$ is randomly generated under the 1st-order statistics under k ready users. The formula is:

$$f_1(y_1)=kf(y_1)(1-F(y_1))^{k-1}$$

For the uniform distribution of f ( ), and normalizing $\beta\equiv\frac{y_1}{T_0}$, we have:

$$f_1(\beta)=k(1-\beta)^{k-1} \qquad (3)$$

(6) $T_{wait,2nd}$ is also generated by the second order-statistics with the condition $T_{wait,2nd}>T_{wait,1st}$. The formula is:

$$f_2(y_2|y_1)=\begin{cases} 0 & \text{if } y_2<y_1 \\ \frac{f_2(y_2)}{1-F_2(y_1)} & \textit{else} \end{cases}$$

where f_2( ) is the 2nd-order statistics under no conditions. By normalizing $\frac{y_2}{T_0}\equiv\gamma$,

$$f_2(\gamma|\beta)=\begin{cases} 0 & \text{if } \gamma\le\beta \\ \frac{k(k-1)\gamma(1-\gamma)^{k-2}}{(1+(k-1)\beta+1)(1-\beta)^{k-1}} & \beta<\gamma\le 1 \end{cases} \qquad (4)$$

(7) By the same explanation as in (4), successful transmission is determined by the condition:

$$\gamma>\beta+\alpha$$

where $\alpha\equiv\frac{T_{pg}}{T_0}$.

(8) During each successful transmission period T_{TR}, or collision period T_{COL}, there are possibly new incoming ready users. The incoming rate is $(N-k)\lambda$, where λ is the individual Poisson access rate.

We assume that whenever a user becomes ready, there is no more new incoming packet, because he is waiting for the response. The value of k is updated based on this process.

(9) The above model is simulated over a long period. The average transmission delay is calculated by the large sample mean.

(10) The value T_0 in the waiting time process can be fixed or adapted. If it is fixed, the access rate $\frac{1}{T_0}$ for each user is fixed. Sometimes it is desired to make the successful access probability fixed and independent of traffic intensity by adapting T_0. The relationship between k, T_0, and the successful access probability is derived as follows.

$$Prob\left\{no\ \ collision \mid \beta\right\} = Prob\left\{T_{wait,1} + T_{pg} < T_{wait,j}\right\} \text{ for } any\ \ j > 2, = k(1-\alpha-\beta)^{k-1}$$

The last equality follows that each other $(k-1)$ ready users has to have waiting time greater than $T_{wait,1st}+T_{pg}$. For successful access probability, we have

$$S \equiv Successful\ \ Probability \equiv \int_0^{1-\alpha} s(T_{wait,1st})d\beta = (1-\alpha)^k \tag{5}$$

Therefore, we have:

$$\alpha = 1 - S^{\frac{1}{k}} \tag{6}$$

A.3: Simulation Results

The simulation results are shown in figures (2,3), where network sizes of 1 Km and 10 Km are compared.

For the case where the successful access probability is constant, it is assumed there is some feedback mechanism in monitoring the traffic to adapt the access rate. By comparing the result of [1], as shown in figure (2), they have about the same saturation point. But by the property modeled in A.2, (8), the simulation result in this chapter shows a much lower slope, and consequently appears more attractive.

For the case where the access rate is fixed, the performance is also dependent upon it. By examining figure (3), an optimum access rate exists which provides a high saturation point. Higher access rates have lower saturation points, because they have a higher collision probability, as shown in figure (5). Lower access rates have lower saturation points, because they have longer average waiting times, which will accumulate more incoming ready users and increase the collision probability.

Both of the cases above seem much more attractive than TSS. These results support the integration of TSS and CSMA.

REFERENCE:

[1]S. S. Lam, *A carrier Sense Multiple Access Protocol for Local Networks*, **Computer Networks, 1980, pp21-32.**

[2]F. A. Tobagi and V. B. Hunt, *Performance Analysis of Carrier Sense Multiple Access with Collision Detection*, **Computer Networks, 1980 pp245-259.**

[3]M. L. Molle, *Unifications and extensions of the multiple access communications problems*, **UCLA-ENG-8118, Computer Science Dept. UCLA, Ph.D. dissertation, July 1981.**

[4]M. L. Molle and L. Kleinrock, *Virtual time CSMA: why two clocks are better than one*, **IEEE Trans. on Communication, Sep. 1985, pp919-933.**

[5]M. Onoe, Y. Yasuda, and M. Ishizuka, *A Random Access Packet Communication System with Priority Function - Priority Ethernet*, **Proceedings of the National Conference on Information Processing Society, Japan, Aug. 1978, 3A-1.**

[6]F. A. Tobagi, *Carrier Sense Multiple Access with Message-Based Priority Functions*, **IEEE Trans. on Communications, Jan. 1982, pp. 185-200.**

[7] Timonthy A. Gonsalves, *Packet - Voice Communication on an Ethernet Local Computer Network: an Experimental Study*, **XEROX, CSL-82-5, March 1982.**

[8] Ming-Kang Liu, David G. Messerschmitt, and David A. Hodges, *Time Slot Switching and Fiber*

Optics for Integrated Service of PBX/LAN, to be published.

[9]Ronald V. Schmidt, Eric G. Rawson, *Fibernet II: A Fiber Optic Local Area Network with Data Collision Sensing*, SPIE 1982, Vol. 355, pp120-126.

[10]J. W. Reedy and J. R . Jones, *Methods of Collision Detection In Fiber Optic CSMA/CD Networks* IEEE Journal on Selected Areas in Communications, Nov. 1985, pp890-896.

[11]K. Oguchi and K. Nosu, *A study on data collision detection circuit for optical star networks*, Instit. Electron. Commun. Eng. Japan-Nat Conference Opt. Electromagnetic Wave, no. 341, Aug. 1982.

[12]Toshifumi Tamura, et al. *Optical Cascade Star Network - A New configuration for a Passive Distribution System with Optical Collision Detection Capability* IEEE Journal of Lightwave Technology, Feb. 1984, pp61-66.

[13]John Detreville, *A Simulation Based Comparison of Voice Transmission on CSMA/CD and Token Buses*, AT&T Bell Lab. Technical Journal, Jan. 1984, pp. 33-55.

[14]D. Bertsekas and R. Gallager, *Queuing Networks*, Prentice Hall, 1986, Chapter 3.

[15]R. B. Cooper and G. Murray, *Queues Served in Cyclic Order*, Bell System Technical Journal, March 1968, pp. 675-689.

[16]R. B. Cooper, *Queues Served in Cyclic Order: Waiting Times*, Bell System Technical Journal, March 1970, pp. 399-413.

[17]P. J. Kuehn, *Multiqueue System with Nonexhaustive Cyclic Service*, Bell System Technical Journal, March 1979, pp. 671-698.

[18] Robert M. Metcalfe and David R. Boggs, *Distributed Packet Switching for Local Computer Networks*, Communications of ACM, July 1976, pp. 395-404.

[19] J. D. Markov and N. C. Strole, *Token-Ring Local Area Networks: A Perspective,* **Proceedings of COMPCON F82, pp. 606-614.**

[20] N. Abramson, *The ALOHA System - Another Alternative for Computer Communications,* **AFIPS Conference Proceedings, Vol. 37, 1970, pp. 281-285.**

CHAPTER 4

Skew Time Slot Switching and Slotted-Ring in a Metropolitan Area Network

4.1. Introduction

In chapters 2 and 3, time slot switching (TSS) and a combination of it with CSMA/CD were proposed to serve all voice, data, and video in a local environment such as LAN by fiber optics and high speed CMOS crosspoint switches [1,2]. With these local area networks (LAN) becoming more prevalent, a larger network known as metropolitan area network (MAN) [3-5] becomes more and more important to support their increasing intercommunications.

As the network size gets larger, propagation delay and signal degradation become more significant problems; as a result, techniques used in LANs cannot be applied directly to MANs. From a system point of view, the major difference between LANs and MANs is the propagation delay. In general, the larger the propagation delay, the larger the "asynchronousness" between users. This asynchronousness reveals itself in different forms depending on medium access protocols. For example, it is the collision detection time in CSMA/CD [6], the token switchover time in a token-ring [7], and the guard time in TDMA or TSS [1,8]. This effect results in less network utilization or longer transmission delays.

For a MAN that is 50 Km in extent, protocols like CSMA/CD or TSS will be impractical and inefficient. For TSS, we have shown the average delay will be proportional to the total guard time (Eq. 15, chapter 2), which is in turn proportional to the propagation delay. For CSMA/CD, the large propagation delay causes not only a large average queueing delay but also a small achievable utilization due to collision [12]. In [12], Lam showed that one of the dominant terms in the queueing delay is proportional to

$$\frac{\eta T_{pg}}{1-\frac{\eta}{x/(x+T_{pg}/S)}}$$

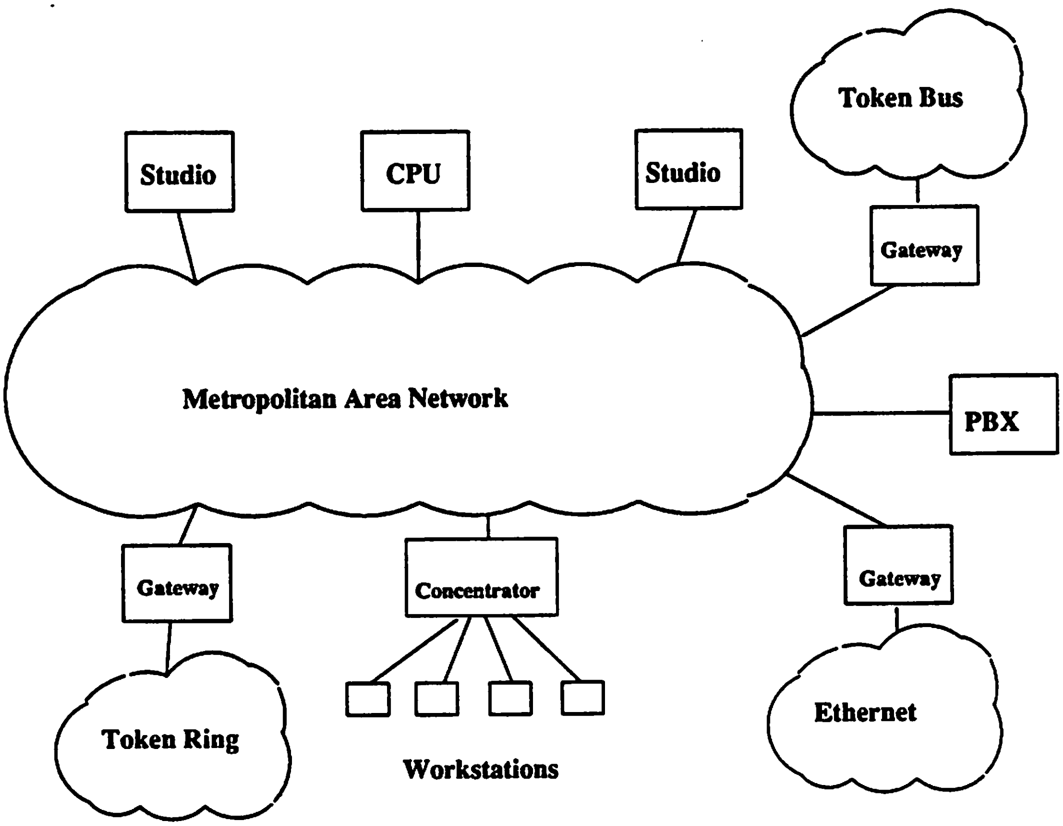
where η is the utilization, T_{pg} is the maximal propagation delay in the network, x is the packet duration, and S is the successful access probability. From the equation, the maximal achievable utilization is

$\frac{x}{x+T_{pg}/S}$, very small when the packet length is much smaller than the network propagation delay.

A token-ring or slotted-ring has similar properties except that the utilization can approach 100% [7,11]. This higher utilization is achievable primarily because adjacent nodes which are interconnected by a physical ring may still be very close even when a larger size network is concerned, if the number of the nodes in the ring is large. Another reason is that when traffic increases, the token holding time will be large compared to the token passing time, which is about the propagation delay. This makes the overhead in token passing relatively insignificant to the information transmission. Consequently, all proposed approaches to MANs are in a ring topology [3-5].

Nonetheless, for the token-ring the average queueing delay is still proportional to the total round-trip delay. If a logical ring with a physical star topology is used, this property will be very undesirable. In

Figure 1. An application of a metropolitan area network (MAN)



addition, there are two major disadvantages to a ring network. First, it requires a very high reliability at each node and link to prevent any possible network failure. A double ring approach [3] is proposed, but the implementation cost is also doubled or more. Second, every link has to support all the traffic generated; that is, network utilization is within 100%. If this network is also to provide video service using 50 to 140 Mb/s channels, not much space will be left for data or voice traffic from a few hundred Mb/s links.

Although the propagation delay is an important factor in larger area networks, one observation does create better opportunities for MANs. As pointed out in chapter 1, a MAN in general provides intercommunication between smaller networks such as LANs, PBXs, large frame computers, and video studios, figure 1. As a result, the network topology can be much simpler than that of LANs. For example, to use TSS for the network in figure 1, a central switching star will be sufficient, figure 2. Every subnetwork, PBX, studio, etc. will exchange their information at preassigned time slots; these assignments and the slot timing will be the same as that explained in chapter 2. This topological simplicity will be utilized to extend the TSS suggested in chapter 2 to larger networks such as MAN.

In the following, two medium access protocols are first proposed: one is called Skew Time Slot Switching (STSS), a modification of TSS, and the other is modified slotted-ring. The STSS is primarily circuit switching and very useful for real-time transmission such as voice or video, while the proposed slotted-ring efficiently handles data packets. As a result, an integration of these two protocols will provide the best service.

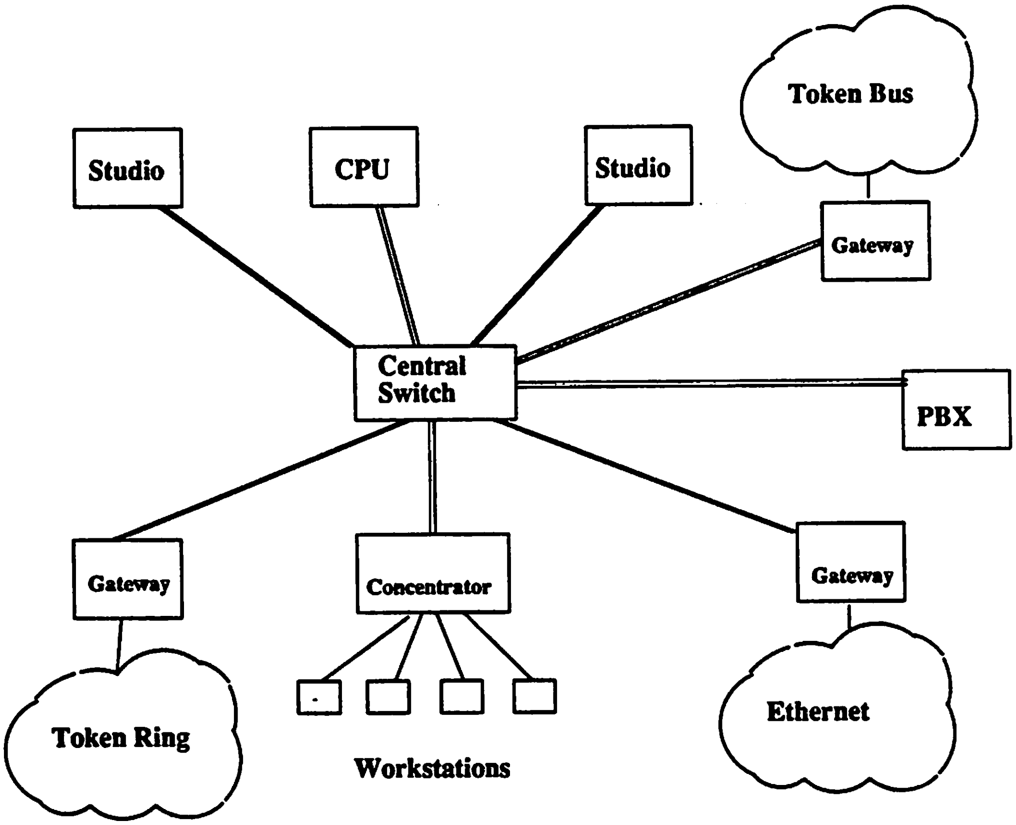
4.2. Skew Time Slot Switching

As pointed out in chapter 2, TSS has large throughput and high reliability, but the guard time inefficiency limits the network size to within 10 Km. To overcome this problem and still preserve its advantages, Skew Time Slot Switching (STSS) is presented.

4.2.1. Basic Concepts

Since the guard time is from: (1) the propagation delay and (2) the global slot timing asynchronousness between switching nodes, these two factors must be reduced to minimize the guard time effect. In

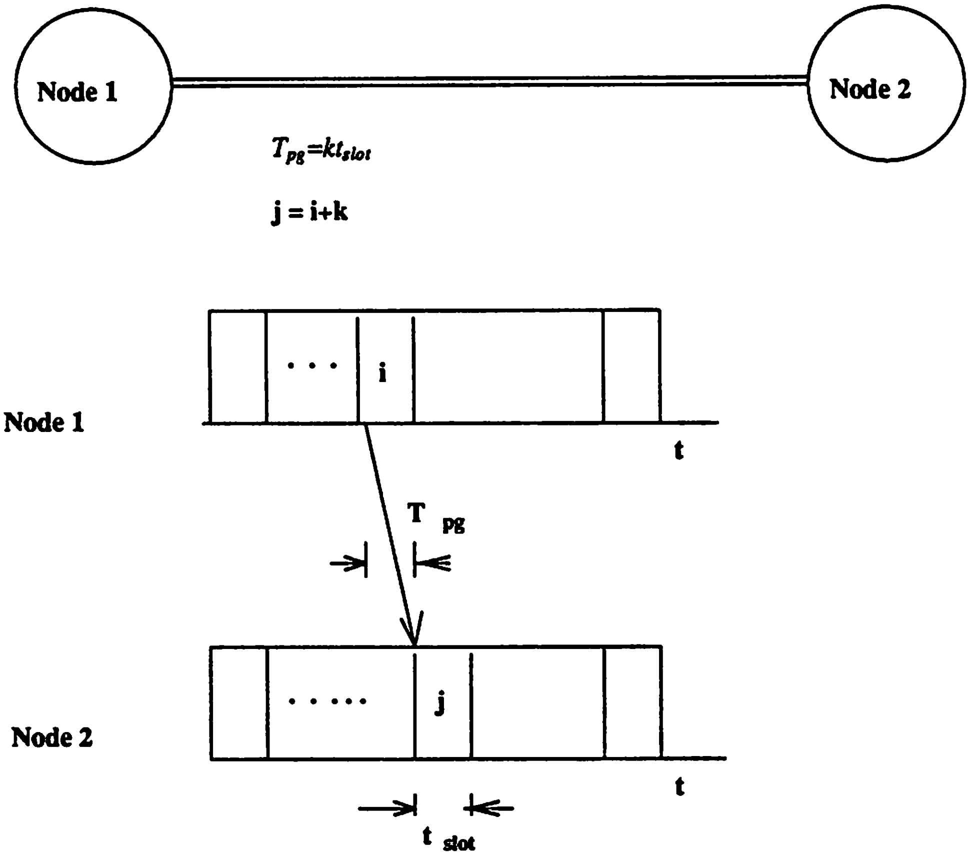
Figure 2. Time Slot Switching for the intercommunication in Figure 1



chapter 2, the global timing skew is minimized by initially measuring the propagation delay between the central controller and the switching node to offset the received global timing.

The effect of the propagation delay on the guard time can also be avoided by using a "skew slot scheduling approach". In this scheme: the "propagation delay" in every link is maintained to be an integer number of slots; under this condition, a packet in the ith slot from one node will arrive at another node during the $(i+k)$th slot, figure 3, where kt_{slot} is the propagation delay through the link. A skew schedule can be defined just like in TSS except that the time slot assigned in each node for the same traffic will be shifted by an integer number of time slots which is equal to the propagation delay.

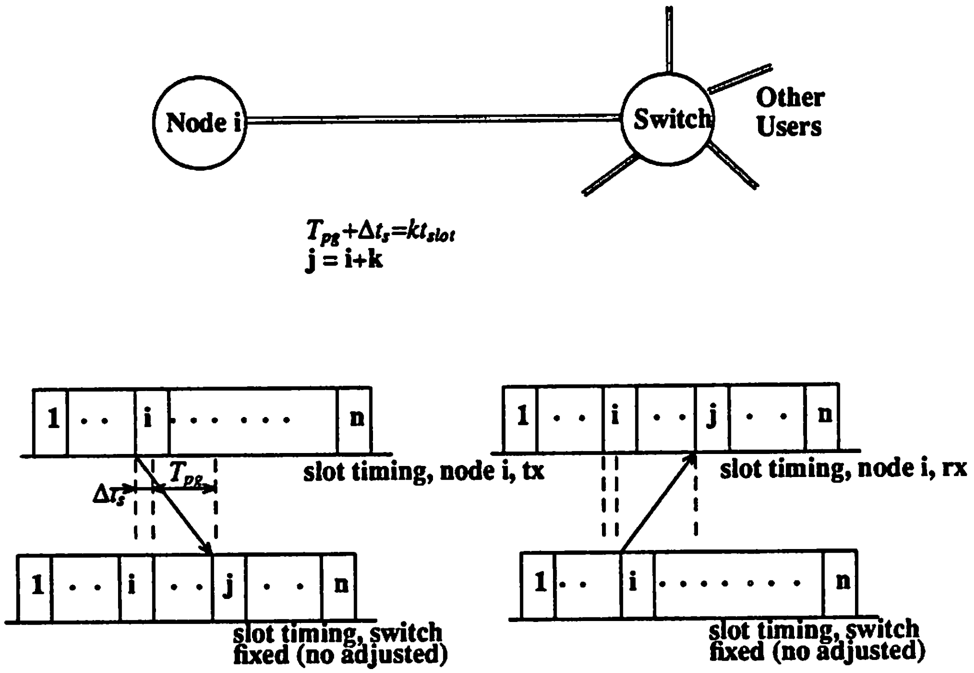
Figure 3. Adjust the propagation delay to minimize the guard time



4.2.2. Practical Implementation

In a practical implementation, the physical propagation delay can be adjusted by the fiber length. For example, with a propagation delay of 5 µsec per kilometer, the imprecision in controlling the delay can be within 50 nsec if fiber lengths can be adjusted within 10 meters. Electronic delay lines can be further used to adjust this delay. However, if the network topology is simple, such as the single star in figure 2, every these delays can be saved by modifying the slot timing. As shown in figure 4, a slot timing with skew Δt_s for one node is intentionally introduced so that Δt_s+T_{pg} in figure 4 is the desired "propagation delay" (i.e. it will be an integer number of time slots). Note that the slot timing for transmission and receiving will be different: they are ahead and behind the proper one by Δt_s, respectively. Also, remember Δt_s can be estimated initially just as the timing skew treated in chapter 2.

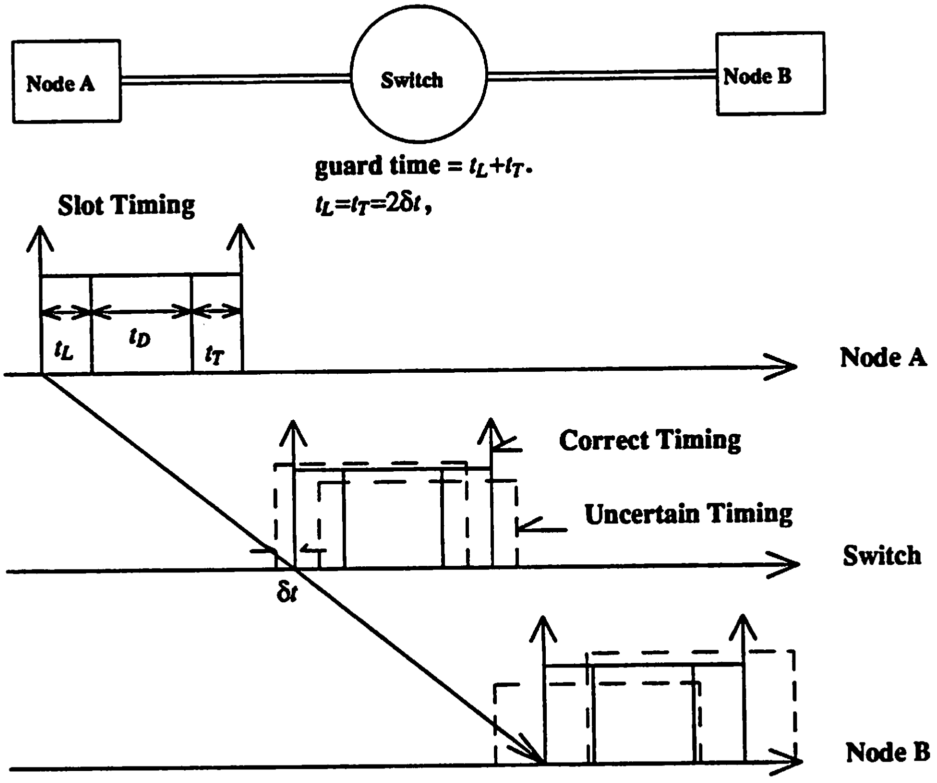
Figure 4. Adjust the slot timing to make the propagation delay as an integer number of time slots in a single star.



As an example, consider the network illustrated in figure 2. With the well-adjusted propagation delay or slot timing mentioned above, say that the propagation delay between the Ethernet and the central switch is k_1t_{slot}, and the delay between the token-ring net and the switch is k_2t_{slot}. As a result, if a packet is sent from the Ethernet to the token-ring net during slot i, it will arrive at the central switch during the $(k_1+i)mod(n)$th slot, and eventually arrive at the token-ring net during slot $(k_1+k_2+i)mod(n)$, where n is the number of slots in a frame.

With the techniques mentioned, both the timing skew and the propagation delay will not be a constituent of the guard time; consequently, the guard time in TSS is only limited to the implementation inaccuracy. This inaccuracy will be very small and insensitive to the network size. The relationship between this inaccuracy and the guard time will depend on the network topology. For a single star network such as shown in figure 2, the guard time will be four times the slot timing inaccuracy, figure 5. With a guard time of 5 μsec, the performance is shown in figure 6.

Figure 5. Relationship between timing inaccuracy with the guard time for a single star.



guard time = t_L+t_T.
$t_L=t_T=2\delta t$,

As mentioned at the beginning of this chapter, a single star topology will generally be sufficient since the topology in MAN applications will be much simpler than in LAN applications. Therefore, by modifying the slot timing, the STSS will introduce no extra implementation and no degradation in performance. If the network has a more complicated structure than a central switching star, the STSS method still can be used, except that delay lines between switching nodes have to be used. As illustrated in figure 7, either the fiber length between the two switching nodes is adjusted to be an integer number of time slots, or a delay line is used in one of the switches. In addition, since each packet passes two switches, the guard time will be six times the timing inaccuracy.

Figure 6. The performance of STSS with guard time of 5 μsec. Each slot is 16 Kb/s, 16 nodes, and at the blocking probability 0.001.

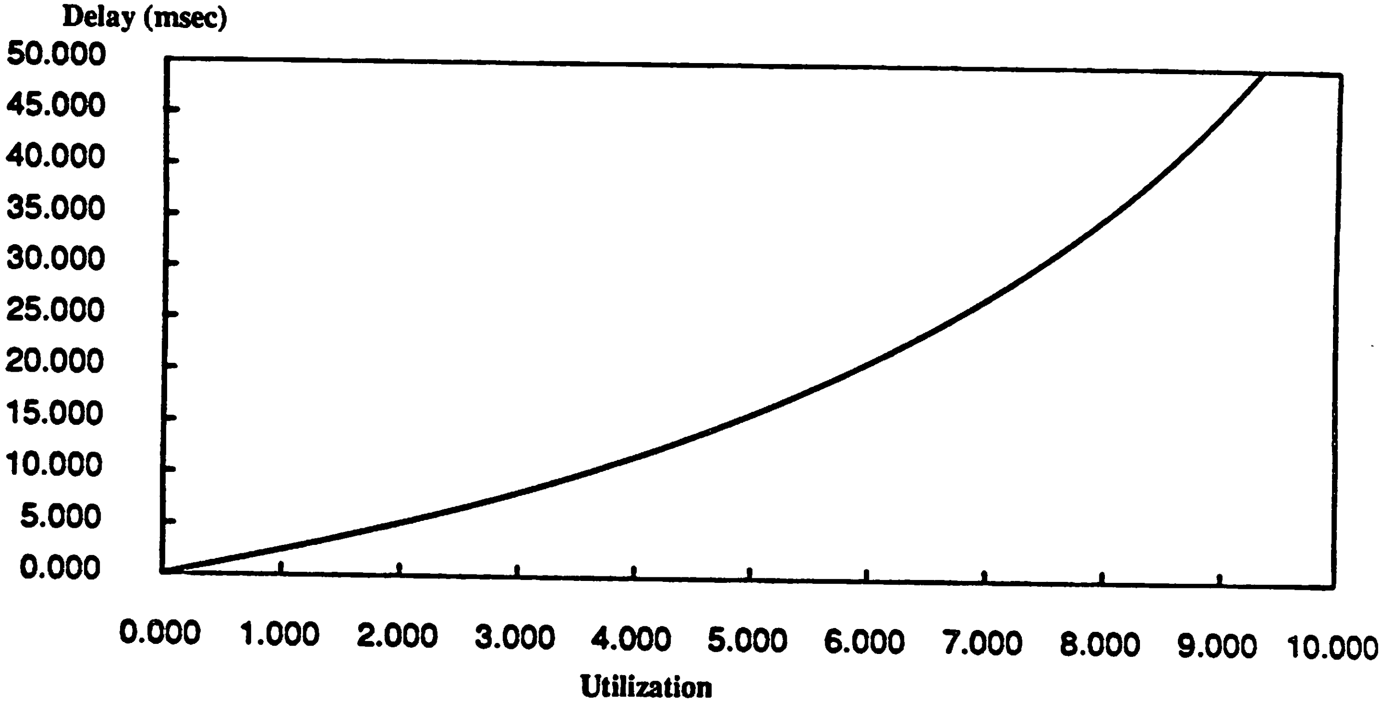
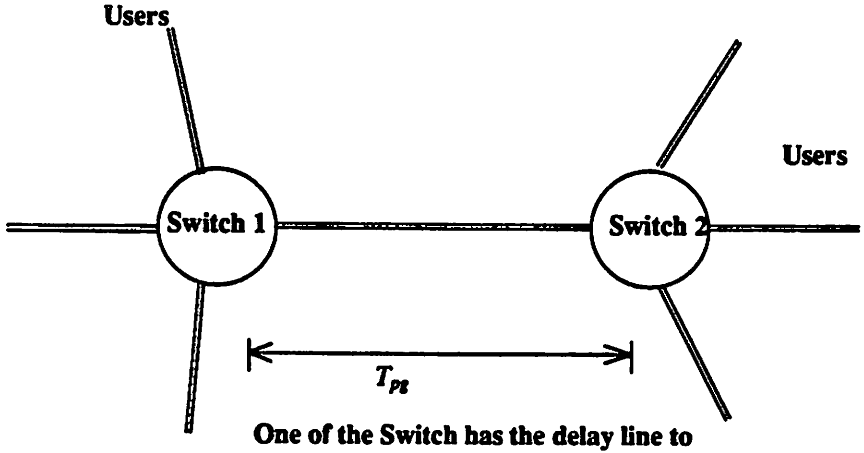


Figure 7. A two switches TSS network



One of the Switch has the delay line to adjust the delay T_{pt} to be an integer of time slots

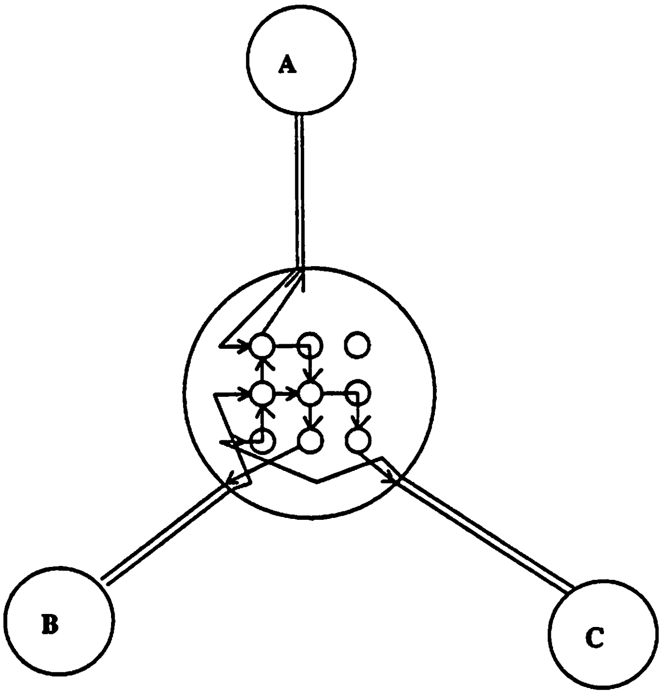
Users adjust their slot timing based on the techniques mentioned above.

4.3. The Slotted-Ring

For a slotted-ring in a star topology, a logical ring is first formed by connecting the active switch as shown in figure 8. Because the physical distance between two adjacent nodes is proportional to the network size, the total round-trip delay around this logical ring will be significantly large. For token-ring, performance analyses have shown that the average transmission delay is largely proportional to this delay [10,11]; therefore, we consider a slotted-ring, which has transmission delay independent of the network size.

In the IEEE 802.6 [5] standard, a frame of an integer multiple of 125 μsec consisting of many time slots is used, so that voice traffic has a fixed delay. Now, we consider there are m slots uniformly distributed and of same length in a logical ring of n nodes. Each slot can be either empty or full. All these slots constantly travel along the ring, and empty slots can be loaded with data information when they arrive at some nodes with data to send. If a full slot arrives at its destination, it can become empty if the destination

Figure 8. A logic ring by a active circuit switch



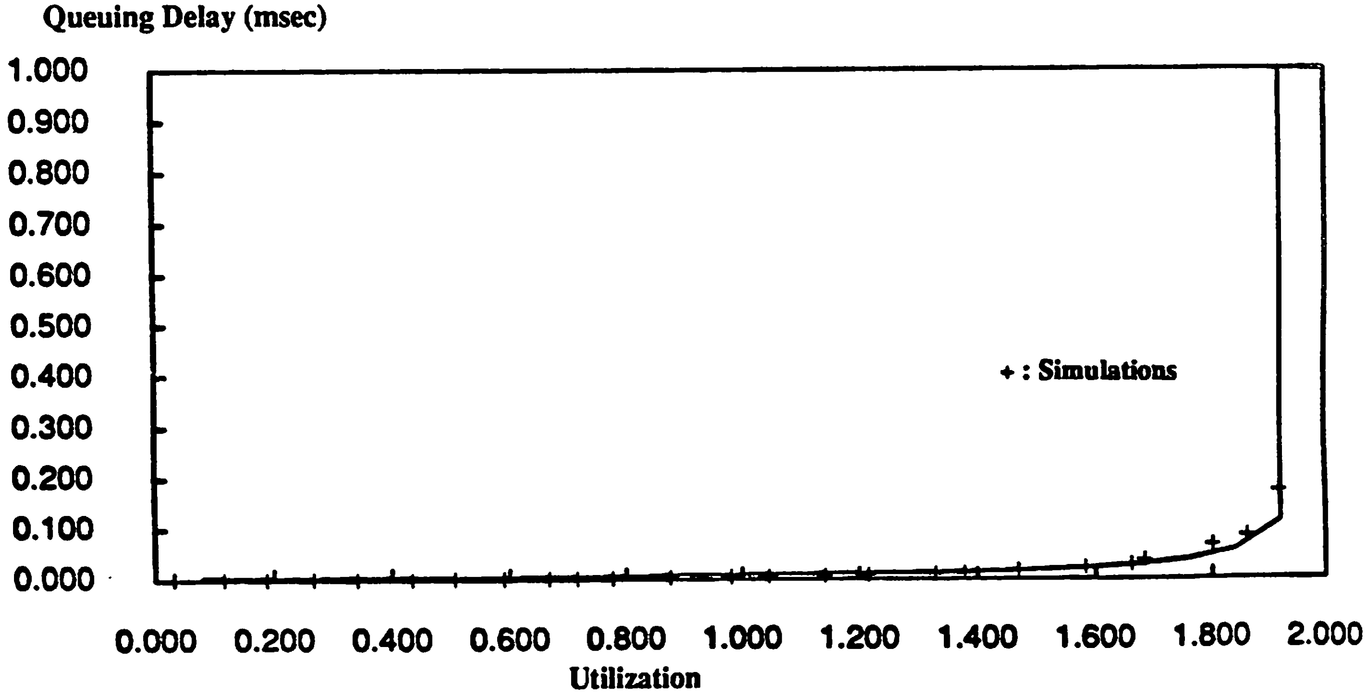
A logic ring is formed from A to B to C and to A.

node has no data to transmit; otherwise, the slot can be immediately loaded by a new packet in the destination node. This approach allows the network utilization to approach 200%, as demonstrated in the analysis and simulation results.

Generally speaking, data packet transmission requires acknowledgements [3,4]. For token-passing, the acknowledgement is automatically achieved by setting a confirmation bit in the packet at the destination node, and the sender will recognize correct transmission by checking the bit when the packet comes back. The disadvantage is that total utilization is limited to 100% or less. However, two nodes often communicate to each other at the same time; in this case, the acknowledgement can be contained in the reply message to avoid extra traffic loads. This is another reason for the proposed slotted-ring.

The analytic derivation for the performance is shown in the Appendix. Some performance is shown in figures 9,10. Since a full slot may be reloaded when it arrives at the destination, and the average transit

Figure 9. The performance of the slotted-ring described with 10 nodes, the propagation delay between adjacent nodes is 5 μsec (about 1 Km). In Figures 9,10,13-16, each slot length is 5 μsec, and the number of slots is same as the number of nodes.



path is half of a ring, utilization can approach 200%. The simulation results are also superimposed on the same graphs as a comparison. The consistency of the simulation and analysis can be observed from these figures.

The most important advantage of this slotted-ring protocol is that the queueing time is independent of the network size. Therefore, a high utilization can be achieved at a reasonable transmission delay even in the case of a large MAN.

4.4. The Integration of Skew TSS and Slotted-Ring

As analyzed in chapter 3 [9], although TSS is very attractive for real-time transmission, some low active interactive data and short data packets will be better served by data packet switching. This motivates the integration of STSS and slotted-ring using the same fiber optics links and active switches in a single MAN.

Figure 10. The same analysis as above except 50 nodes in the ring.

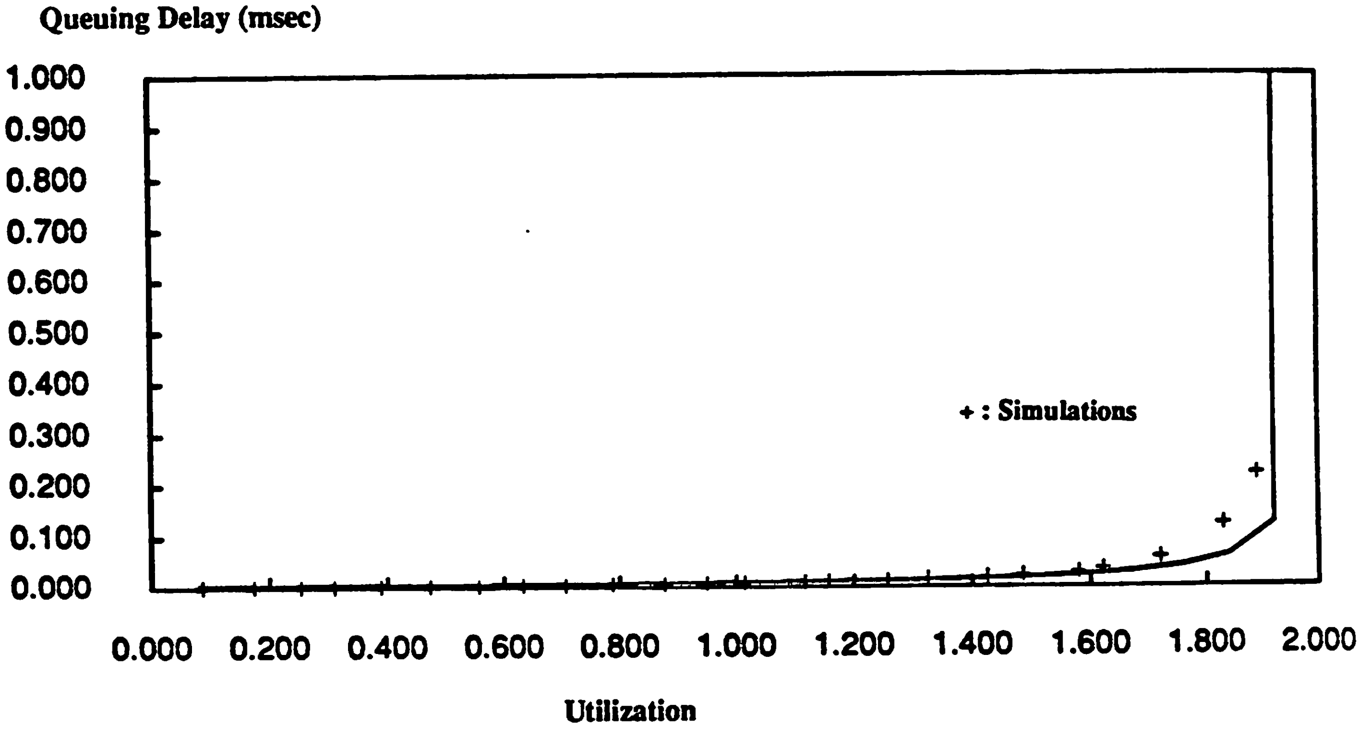
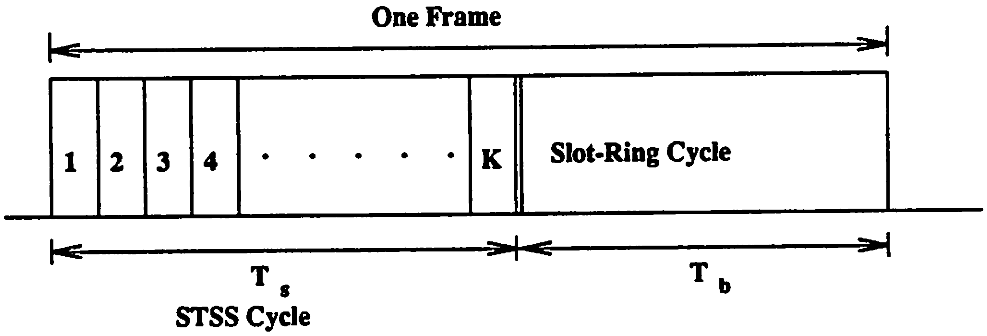


Figure 11. The integration of STSS and the slotted-ring.



The integration is accomplished by dividing a time frame into two subframes: one for STSS and another for slotted-ring, figure (11). In each subframe, the protocol follows those introduced in the previous two sections, and the switches in the slotted-ring subframe have a logical ring configuration. As in TSS, video traffic will be assigned a dedicated link during its conversation (i.e. no time division with other users). Voice traffic will be assigned to STSS slots because STSS provides a fixed transmission delay. Some data traffic of fixed but not small length can also be assigned to STSS slot(s). Because multiple transmissions can be switched simultaneously, communication links are well utilized. Data traffic with short length or of constant rate is assigned to slotted-ring slots. Short packets are not assigned STSS slots because the overhead in slot scheduling is comparatively too long. Constant rate data traffic generally has very low active time; therefore, packet switching will function more efficiently than STSS. These allocations based on traffic characteristics provide not only attractive services but also high network utilization.

Because the propagation delay may be larger than one slot size, a switchover time is required between the two subframes. As shown in figure (12), this time is the maximum propagation delay in the network for STSS, or the delay between two adjacent nodes in the slotted-ring case. The time in these two cases is the same and linearly proportional to the network size. For example, assuming a propagation delay of 5 μsec/Km, it will be 0.25 msec for a 50Km network. If a frame of 50 msec is considered, the inefficiency due to this is only $2 \cdot 0.25/50 = 1\%$.

The compression delay for STSS after the integration will be the same for the pure STSS because of circuit switching characteristics. The performance of the slotted-ring after the integration may have longer

Figure 12. The switchover overhead between STSS and the slotted-ring.

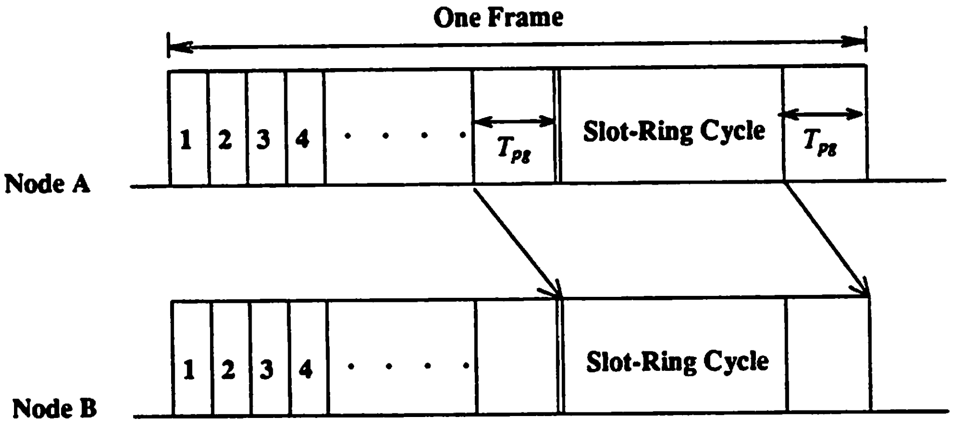


Figure 13. The performance of the integration of STSS and the slotted-ring.
10 nodes in the network and 1 msec frame.

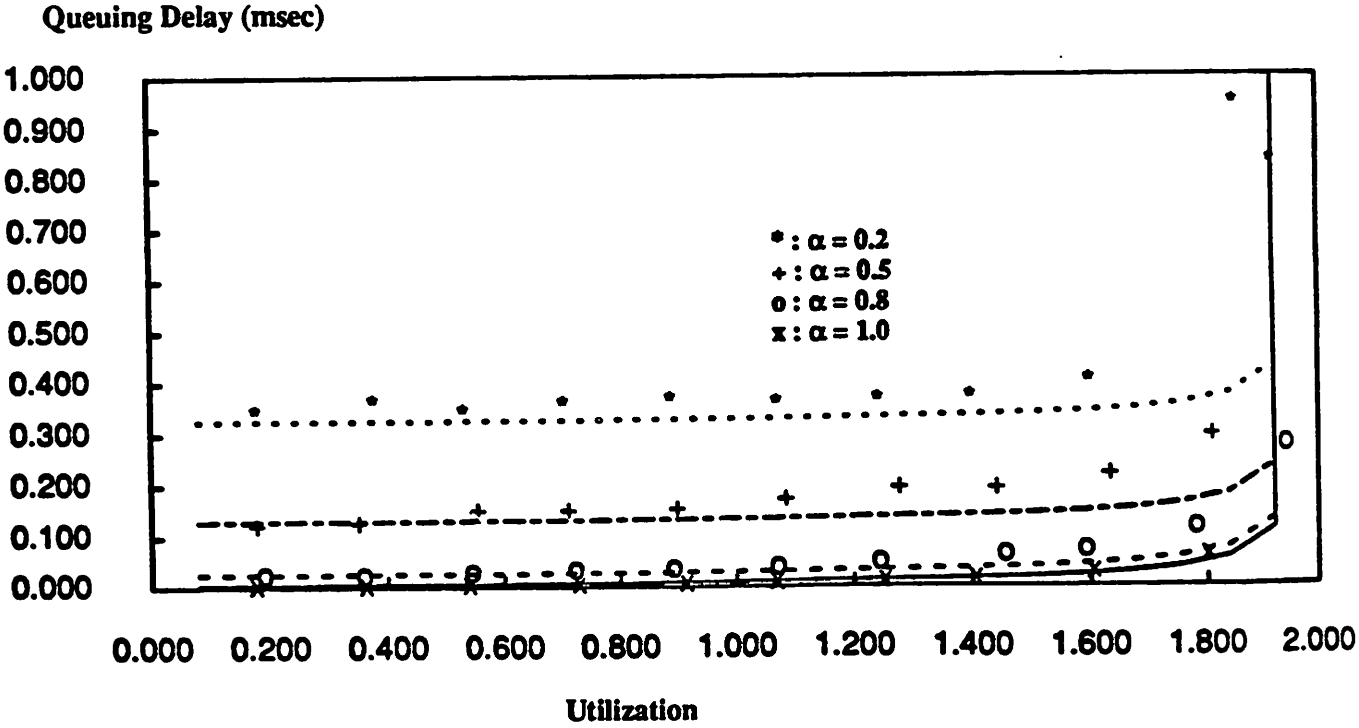
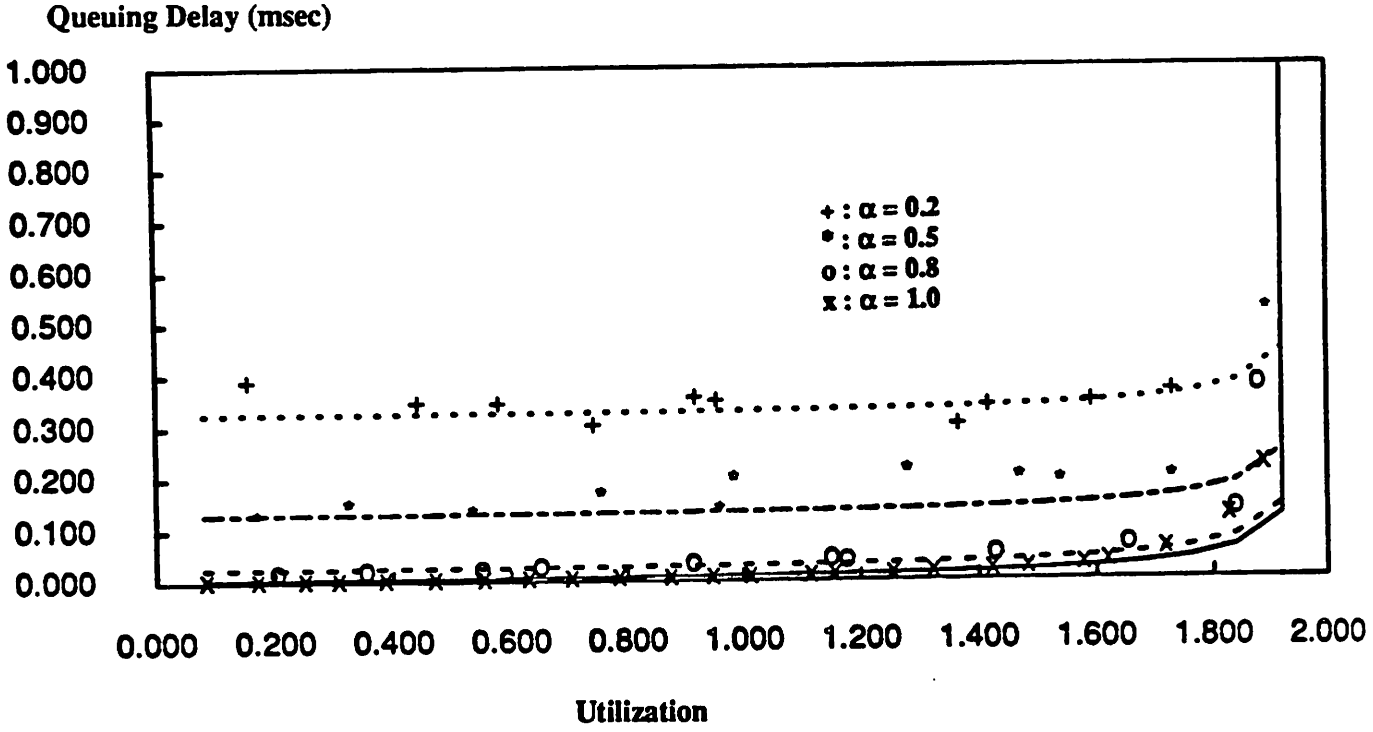


Figure 14. The performance of the integration of STSS and the slotted-ring. 50 nodes in the network and 1 msec frame.

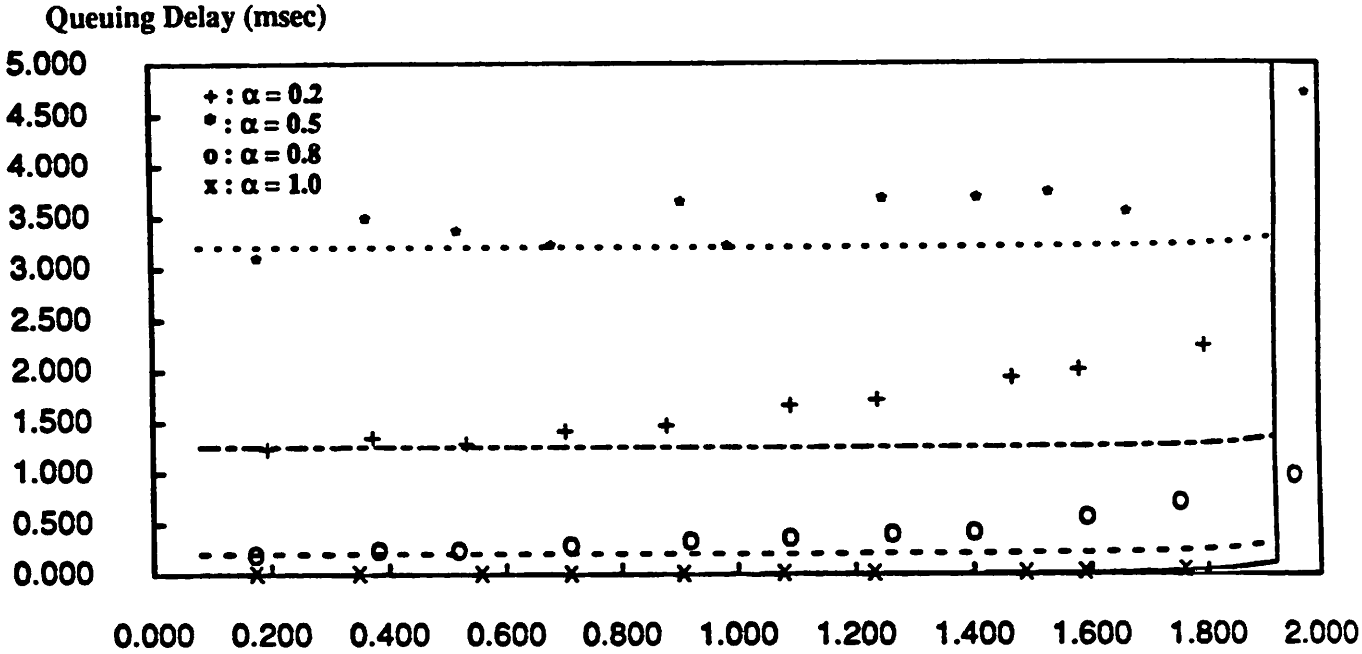


average queueing delay because (1) traffic is not served during STSS subframes, and (2) slots have higher full probability at the beginning of the slotted-ring subframes. The extra queueing delay due to traffic not served in these subframes is

$$T_d = \frac{(1-\alpha)^2}{2} T_{frame}$$

where α is the percentage of the slotted-ring in each frame. The extra delay due to this higher full probability cannot be easily analyzed, but it will be insignificant if traffic is low. Therefore, the performance is analyzed by simulations. The simulation results for different nodes, different α, and different frame times are shown in figures (13-16). The delay time analyzed in the pure slotted-ring case plus the extra delay due to traffic not served in STSS is superimposed in the same graphs. The results show they are very close at low traffic intensity, and the additional delay due to higher full slot probability is not significant at higher traffic intensity.

Figure 15. The performance of the integration of STSS and the slotted-ring. 10 nodes in the network and 10 msec frame.

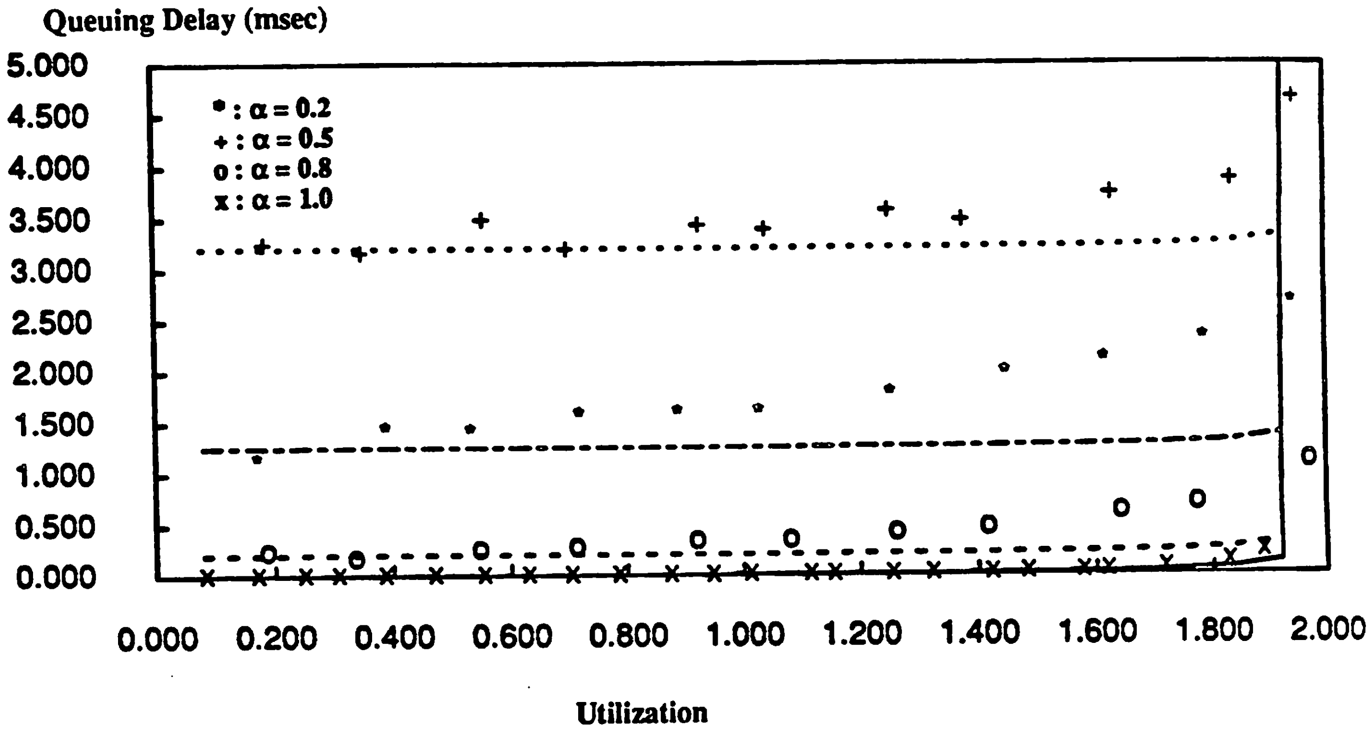


Utilization

4.5. Conclusion

The approach to MAN presented in this chapter not only provides a general service for all voice, data, and video traffic, but also has higher network reliability and flexibility as compared to the FDDI or 802.6 approach. If any node malfunctions, the active switch can just bypass it and a new logical ring still functions. If any new node is added to the system, the system doesn't need to make any adjustment except connecting the new node to the switching node. In addition, by directly connecting a switch between two users for video communication, the bandwidth of each link doesn't need to be very high, and lower speed technologies can be sufficient.

Figure 16. The performance of the integration of STSS and the slotted-ring. 50 nodes in the network and 10 msec frame.



Appendix. Analysis for the Slotted-Ring

In this appendix, the performance of the proposed slotted-ring in section 3 is analyzed. Some network parameters are first defined:

n The total number of nodes in the network.

m The total number of slots around the ring of the network. In this analysis, $m \leq n$ is considered.

λ The Poisson input rate of each node. Assume it is the same for every nodes.

t_p The propagation delay (and possibly plus the slot processing time) between two adjacent nodes. Assume it is same between every pair of adjacent nodes.

t_s The slot length.

η The network utilization, and $\eta = n \lambda t_s$.

T_q The average queueing delay at each node.

T_p The average propagation delay of each packet.

T_{tot} The average transmission delay of each packet, and it is $T_q + T_p$.

Next, let us define the following probabilities:

q_i The probability that i packets are waiting at any node for an available slot to transmit. Therefore, q_0 is the probability that the queue of a node is empty.

P_{arv} The probability that a slot arrives at one node within time period t_s.

P_F The probability that a slot received by any node is full (occupied).

P_E The probability that a slot received is empty. That is: $P_E = 1 - P_F$.

P_{get} The probability for a node to be the destination of a full slot.

P_{avl} The probability for a node to get an accessible slot in t_s. By the above definitions, we have:

$$P_{avl} = (P_F P_{get} + P_E) P_{arv} \qquad <1>$$

Also, at the equilibrium state, the total input packets should be equal to the total output packets. Hence:

$$P_{avl}(1-q_0)=\lambda t_s \qquad <2>$$

In the following, we are going to derive $P_F, P_{get}, P_{avl}, P_{arv}$, and$q_0$. After that, distribution for q_i can be obtained, and the average queueing time T_q follows by Little's Law.

Derivation of P_{arv} and P_{get}

There is an inequality for n, m, t_s, and t_p:

$$mt_s \le nt_p$$

In the following, equality of the above equation is assumed to maximize the network utilization. In this case, each node will receive a slot every t_s period, so

$$P_{arv}=1 \qquad <3>$$

Suppose a full slot will travel k nodes from its source to the destination. If this slot passes a node, what is the probability that this node is the destination? It is just $\frac{1}{k}$. Consequently, assume the probability that a slot will travel k nodes is $P_{pass}(k)$, we have:

$$P_{get}=\sum_{k=1}^{n-1}\frac{P_{pass}(k)}{k}$$

Now the question is what is $P_{pass}(k)$? It has to be proportional to k since the possibility that a node will get the slot will increase with the number of nodes it passes. Hence:

$$P_{pass}(k)=\frac{k}{\sum_{i=1}^{n-1} i}$$

By combining the above two equations, we have:

$$P_{get}=\sum_{k=1}^{n-1}\frac{1}{\frac{1}{2}n(n-1)}=\frac{2}{n} \qquad <4>$$

Derivation of P_F,P_{avl},andq_0

By Eqs. 1,3,4, we have:

$$P_{avl}=1+P_F(\frac{2}{n}-1)$$

There is another relationship for P_{get},P_F,andP_E:

$$P_F=(P_F P_{get}+P_E)(1-q_0)+P_F(1-P_{get}) \qquad \text{<5>}$$

The reasons for the above equation are that (1) P_F is same for each node by symmetry at given same traffic density, and (2) the probability that a slot is full when received by node $(i+1)$ is equal to the sum of two exclusive probabilities: (i) the slot is accessed and full at node i and (ii) it is full before its arrival at node i and not accessible by node i. The equation can be reduced to:

$$P_F P_{get}=P_{avl}(1-q_0)$$

By Eq. 2, we have:

$$P_F=\frac{n\lambda t_s}{2}=\frac{\eta}{2} \qquad \text{<6>}$$

By Eq. 1, P_{avl} is:

$$P_{avl}=1-\lambda t_s\left(\frac{n}{2}-1\right)$$

Combining with Eq. 2, q_0 can be obtained:

$$q_0=\frac{1-\lambda t_s\frac{n}{2}}{1-\lambda t_s\left(\frac{n}{2}-1\right)} \qquad \text{<7>}$$

Derivation of q_i

The remaining problem is to determine the queue size distribution. For each time interval t_s, a recursive expression of q_i is:

$$q_i=\sum_{j+k=i;j\neq 0} q_j p_k(1-P_{avl})+\sum_{j+k=i+1;j\neq 0} q_j p_k P_{avl}+q_0 p_i \qquad \text{<8>}$$

$$=\sum_{j+k=i} q_j p_k(1-P_{avl})+\sum_{j+k=i+1} q_j p_k P_{avl}+q_0 P_{avl}(p_i-p_{i+1})$$

where p_i is the probability that i packets arrive during a period of t_s. For Poisson process, $p_i=e^{-\lambda t_s}\frac{(\lambda t_s)^i}{i!}$.

Taking z-transform: $Q(z)=\sum_{i=0}^{\infty} q_i z^i$,

$$Q(z)=Q(z)P(z)(1-P_{avl})+P_{avl}z^{-1}(Q(z)P(z)-q_0 p_0)+P_{avl}q_0(P(z)-z^{-1}(P(z)-p_0))$$

where: $P(z)$ is the z-transform of the input process, and for Poisson process, $P(z)=e^{\lambda t_s(z-1)}$.

By simple manipulations, we have:

$$Q(z)=q_0\frac{P_{avl}P(z)(1-z^{-1})}{(1-P(z))+P_{avl}P(z)(1-z^{-1})} \qquad <9>$$

The average queue length of each queue is $Q'(1)$, and can be obtained as follows: Let

$$Q(z)=q_0\frac{B(z)}{A(z)+B(z)}$$

where $A(z)=1-P(z)$, and $B(z)=P_{avl}(1-z^{-1})P(z)$. Therefore,

$$Q'(z)=q_0\frac{B'(z)A(z)-B(z)A'(z)}{(A(z)+B(z))^2}$$

But $A(1)=B(1)=0$, by applying L'Hopital's rule twice, and assuming Poisson input process:

$A'(1)=-\lambda t_s$,

$A''(1)=-(\lambda t_s)^2$,

$B'(1)=P_{avl}$,

and $B''(1)=2P_{avl}(\lambda t_s-1)$, incorporating with Eq. 3 we have:

$$\begin{aligned} <q_i> \equiv Q'(1) &= q_0\frac{B''(1)A'(1)-B'(1)A''(1)}{2(A'(1)+B'(1))^2}. \\ &= \frac{1-q_0}{q_0}\left\{1-\frac{P_{avl}}{2}(1-q_0)\right\} \qquad <10> \\ &= \frac{t_s}{1-\lambda t_s\frac{n}{2}}(1-\frac{\lambda t_s}{2}) \end{aligned}$$

Delay vs. Utilization

The final step is to derive the delay vs. utilization relationship. By applying the definition of utilization $\eta=n\lambda t_s$ and Eqs. 7,10,

$$T_q=\frac{t_s}{1-\frac{\eta}{2}}(1-\frac{\eta}{2n}) \qquad <11>$$

The term in the bracket is approximately 1 when n is large, so the first term is the most important and dominant at large traffic intensity.

If each packet has a uniform distribution of sources and destinations, the average propagation delay is $T_p=\frac{nt_p}{2}$, and the total transmission delay will be:

$$T_{tot}=\frac{t_s}{1-\frac{\eta}{2}}(1-\frac{\eta}{2n})+\frac{nt_p}{2} \qquad <12>$$

Eq. 12 gives the analytical solution of the packet transmission delay vs. the network utilization. Some interesting points can be noticed:

(1) If $mt_s=nt_p$, network utilization can approach a maximum of 200%.

(2) At lower traffic intensity, the transmission delay is dominated by the propagation delay. At higher traffic intensity, it is dominated by the queueing delay.

(3) For the queueing delay, the numerator is t_s, which is the slot length, and consequently is independent of the maximum physical propagation delay of the network.

(4) As a numerical example, for a network of 50Km with 5$\mu sec/Km$, t_p is 250 μsec. Assuming also $m=n$ and $t_p=t_s$, a small queueing delay of 2.5 *msec* at 180% utilization can be achieved. On the other hand, with 16 nodes in the network, the average propagation delay will be 2 *msec*.

REFERENCE:

[1] Ming-Kang Liu, David G. Messerschmitt, and David A. Hodges, *Time Slot Switching and Fiber Optics for Integrated Service of PBX/LAN*, to be published.

[2] Ming-Kang Liu, David G. Messerschmitt, and David A. Hodges, *An Approach to Fiber Optics Data/Voice/Video LAN*, INFOCOM 86, pp. 516-523.

[3] Robert W. Klessig, *Overview of Metropolitan Area Networks*, IEEE Communications Magazine, January 1986, pp. 9-15.

[4] Floyd E. Ross, *FDDI - A Tutorial*, IEEE Communications Magazine, May 1986, pp. 10-17.

[5] Draft of Proposed IEEE Standard 802.6, *Metropolitan Area Network (MAN) Media Access Control*, August, 1985, Revision D.

[6] Robert M. Metcalfe, and David R. Boggs, *Ethernet: Distributed Packet Switching for Local Computer Networks*, Comm..of ACM, July 1976, pp. 395-404.

[7] Alan G. Konheim, and Bernd Meister, *Waiting Lines and Times in a System with Polling*, Journal of ACM, July 1974, pp470-490.

[8]. Kamilo Feher, *Digital Communications, Satellite/Earth Station Engineering*, Chapter 8, Prentice-Hall press, 1983.

[9] Ming-Kang Liu, and David G. Messerschmitt, *Time Slot Switching and CSMA in Fiber Optics LAN for Integrated Service*, to be submitted.

[10] L. F. Moraes and I. Rubin, *Analysis and Comparison of Message Queuing Delays in Token-Rings and Token-Buses Local Area Networks*, Proceedings of ICC 84, pp. 130-135.

[11] W. Bux, *Local Area Subnetworks: A Performance Comparison*, IFIP Proceedings, 1981, pp. 157-181.

[12] Simon S. Lam, *A Carrier Sense Multiple Access Protocol for Local Networks*, Computer Networks, February 1980, pp. 21-32.

Part II.

High Speed Network Implementation

CHAPTER 5

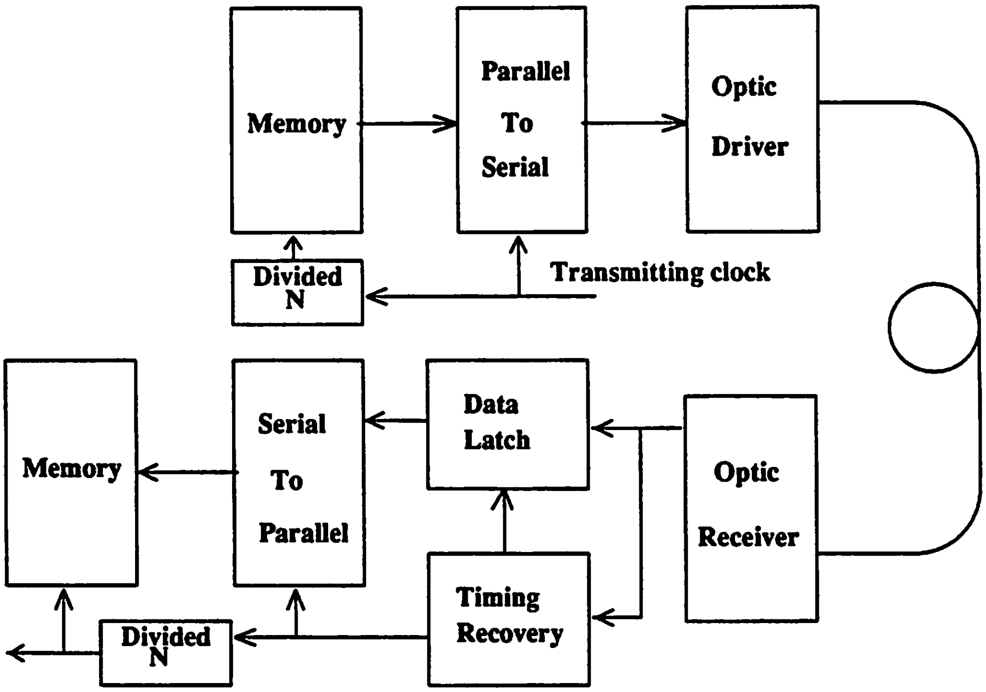
A Fixed Transition Coding for High Speed Timing Recovery

5.1. Introduction

In chapters 2 to 4, we have described several network topologies and access protocols for users to share a common network. The following three chapters will discuss some practical considerations in network implementation. As mentioned in the previous chapters, a TSS network needs transmission links, circuit switches, and a central controller. For high speed transmission, timing recovery is a critical aspect of the receiver design. This chapter addresses this problem and is oriented in particular towards low cost realization. The high speed circuit switch has been proposed and designed separately [10] and will not be described in this thesis. The absense of timing recovery in TSS intermediate switching nodes has been suggested in chapter 2 and needs further experimental justification. For example, the timing jitter may accumulate after several nodes in cascade and become sufficient to degrade the transmission quality. The next chapter will focus on this problem. To implement a whole TSS network, a central controller and other control functions in the switching nodes are also necessary, as described in chapter 2. Although no TSS network has been implemented as yet, realization of these control components will be discussed in chapter 7.

Timing recovery is a necessary function in every digital receiver that provides a synchronous clock to latch the received bit streams. This recovery is traditionally achieved by either a phase lock loop (PLL) or a narrow band pass filter (e.g. the SAW filter). For fiber optics applications in the range from a few hundred Mb/s to a few Gb/s, both the PLL and SAW filter have the problem that the voltage control oscillator (VCO) and the phase detector in the PLL or a high quality factor (Q) in the filter is difficult to implement By examining the characteristics of the communication networks, the electronics interfaces, and the fiber optics more carefully, other simple alternatives do exist for timing recovery.

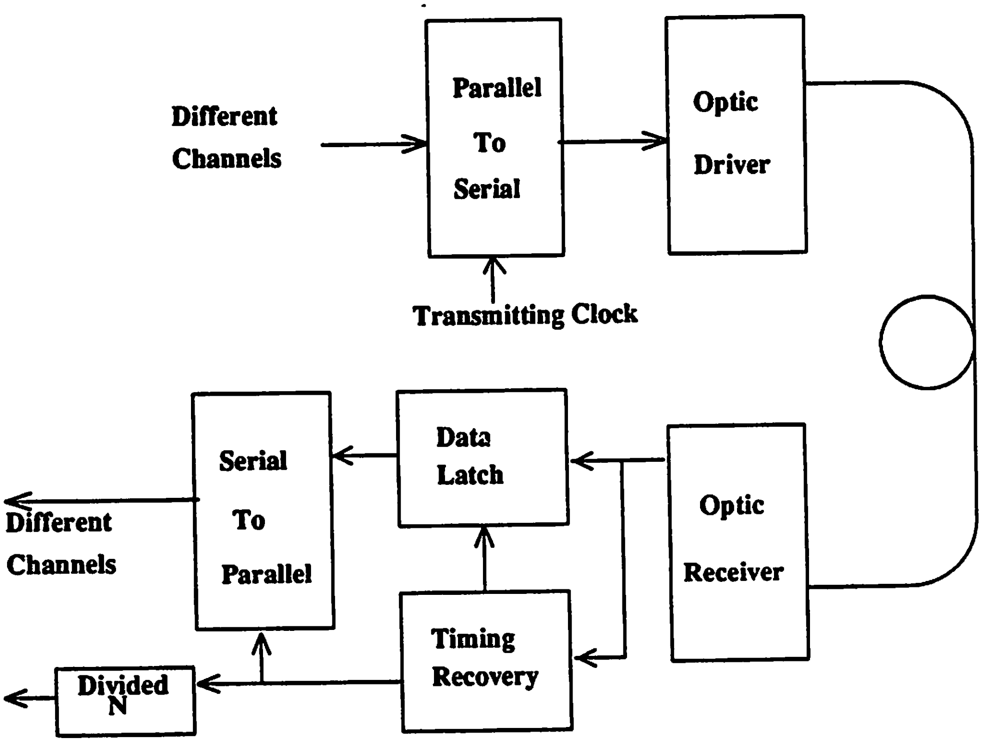
Figure 1. A message interleaving system.



The Large Bandwidth Networks. In practical fiber optic local area networks (LAN) or metropolitan area networks (MAN), users with low speed traffic share a large bandwidth network. The bandwidth is so large that no single message will require the whole bandwidth for a long time. The messages are commonly transmitted by time division (e.g. TDM), spatial division (e.g. circuit switches), frequence division (e.g. FDM or WDM), or a combination of them (e.g. Time Slot Switching in [1]). Of these, some variation of time division is most common in communication networks. For example, packet broadcasting in CSMA/CD networks (Carrier Sense Multiple Access with Collision Detection), token passing in the bus or ring networks, T-carrier systems in telephone networks, and TDMA (Time Division Multiple Access) in satellite networks all belong to this category [2-4]. In time division, there are two subclasses. In message interleaving (Figure 1), each message is transmitted as a packet for a very short period but at the full link transmission rate. Both the transmitter and receiver have a memory buffer to store the message. CSMA, token-passing, TDMA, and TSS are examples of this subclass. For bit/byte interleaving (Figure 2), a fixed number of channels are interleaved by bits or bytes synchronously in the transmitter, and demultiplexed at the receiver. T-carrier is one example. As a result, not only bit timing recovery is required, but there is also a synchronous demultiplexing and framing function. The conversion process provides: (1) speed matching between the channels and users, (2) message formating for the high level information processing (in bytes or words), and/or (3) demultiplexing of the bit streams into different channels (bit interleaving time division). This suggests the bit timing may not be necessary if only the "word" or "byte" timing is eventually required.

The "Low" Speed Electronics Interface. The electronics interface serves as a vehicle between the optical channels and users. As compared to the available transmission rate provided by fiber optics, the speed of mature low-cost commercial electronics is quite low. For example, standard ECL gates [5] have a switching time in the order of one nsec. After including practical wiring and layout problems, and adding a safety margin, word rates higher than the order of 100 Mb/s cannot be processed. If we use parallel processing of words, the bit rates can be much higher. However, as shown

Figure 2. A Bit/Byte Interleaving System.



in figures 1 and 2, electronics which recovers the bit timing at the full link rate is inevitable. Given a bandwidth capability of a few Gb/s by fiber optics [6], the available low-cost electronics technologies represent a bottleneck. The speed is also limited by the higher level processing needs and the memory access time, which are presently in the range of 10 nsec to a few hundred nsec. These considerations suggest that parallel processing will provide higher effective bandwidths for a given electronics technology.

Fiber Optic Channels Characteristics. Fortunately, the characteristics of the fiber optics channel encourage simple timing recovery techniques. Fiber optics provides a medium which has a high SNR (particularly in LAN and MAN applications) and low dispersion. For example, a bit error rate of 10^{-12} at -15 dbm, 100Mb/s, and 10 Km using LED/PIN and multi-mode fibers has been reported [7]. A higher transmission rate at 1.6 Gb/s, 10 Km, and -18 dbm by single mode fiber and LD/APD with bit error rate of 10^{-11} has also been achieved [8]. With this high quality medium, a simple coding technique enhancing the timing information can be used to facilitate timing recovery in the receiver. Furthermore, since the medium has almost unlimited bandwidth in relation to the electronics speeds, we can consider introducing inefficiency in bandwidth utilization if there is a payoff in a higher effective speed in the electronics.

The Approach. In this chapter, we propose a line coding and parallel processing technique to reduce the electronics speed required to implement the the timing recovery function. The line coding uses bandwidth inefficiently, since the effective data rate is lower than the actual bit rate. In section 2, the concept of a fixed transition block code is introduced. By dividing the incoming data by a number equal to the number of transitions in each block, a regular block timing is obtained without using any PLL or BPF. Parallel data latching using a delay line recovers all the information bits within a block. The technical issues, including the block synchronization, coding efficiency and conversion, and the implementation of the delay line are discussed in detail in section 3. Experiments using standard ECL gates at 50 Mb/s and 200 Mb/s are discussed in section 4.

5.2. The Block Timing Recovery

The key ideas in our timing recovery approach are to encode the timing information with data information in the choice of the line code, and to shift the high speed operation to a lower speed region. We accomplish this by using a block coding algorithm which introduces a fixed number of transitions within a known block of data symbols, and generating a square wave timing clock at the receiver by simply counting transitions. Other phases of the timing, required for detection of the individual bits within a block, can be generated with a delay line, generating a parallel representation of the information bits as often desired in accordance with system requirements. The technique does not require a PLL or bandpass filter. We illustrate it below with a simple example.

5.2.1. A Fixed Transition Block Code

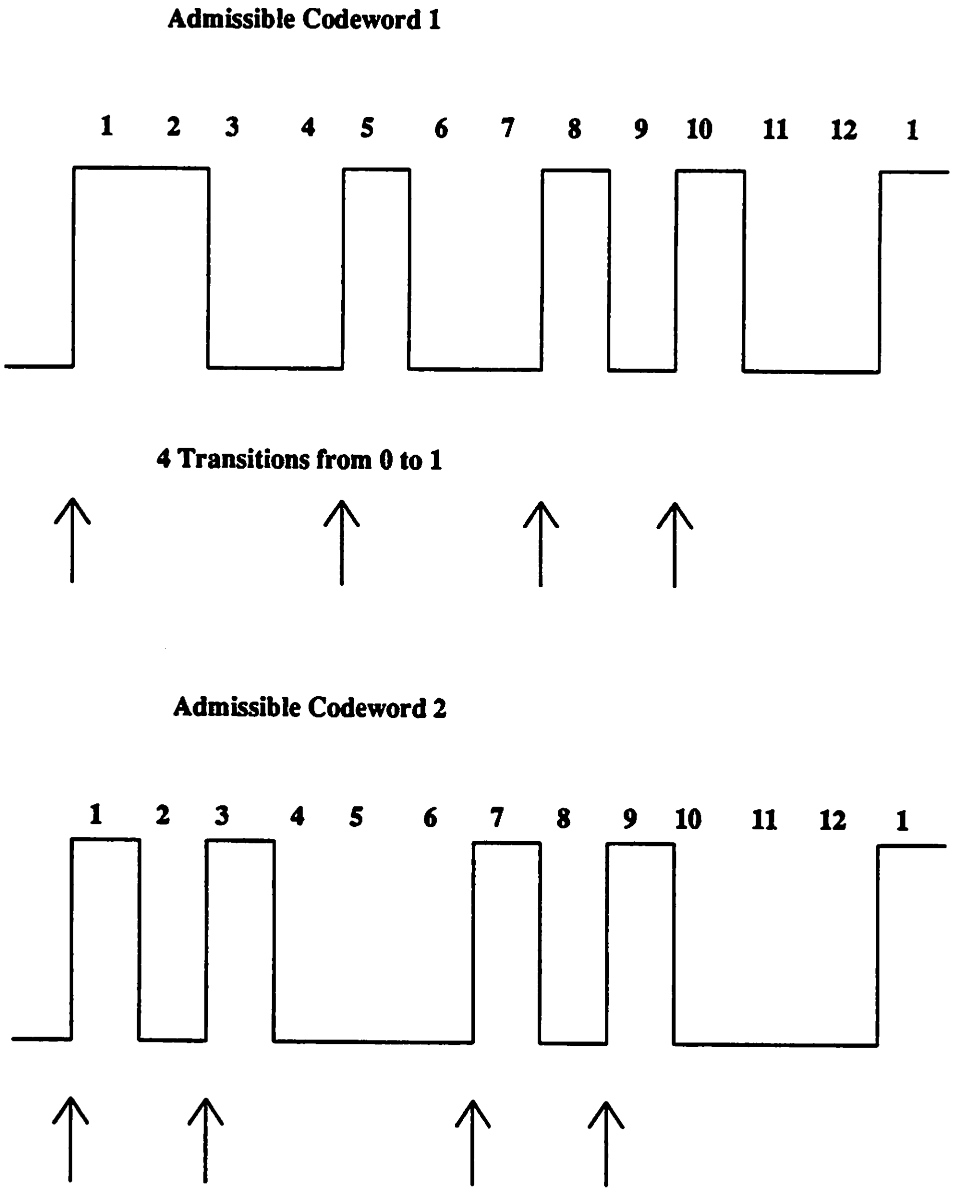
The fixed transition block code has the following properties to carry both timing and data information: (1) in each encoded block of n bits, the number of the transitions from 0 to 1 is fixed, say k; (2) The first transition in each block always occurs between the last bit of the previous block and the first bit of the block. In other words the first and last bits of each block are 1 and 0 respectively.

For example, let each block have n=12 bits and a number of transitions k=4. Two sample encoded blocks are shown in figure 3 for two admissible codewords. Later, we will determine the number of information bits per block, which is related to the number of admissible codewords that meet the constraints.

5.2.2. Timing Recovery

The key problem in timing recovery is deriving a data clock in spite of the random nature of transitions in the data stream. For the fixed transition code this problem is mitigated, as long as we are willing to derive a *block timing* waveform, since the number of transitions is fixed regardless of the data. The block timing can be recovered by simply using a divide-by-k counter that divides the number of transitions by k. If the phase of the countdown in correct, the output waveform will be a square wave synchronous with the bit timing, but at a frequency $2n$ times lower. By exclusive-or'ing

Figure 3. Illustrations of admissible codewords.

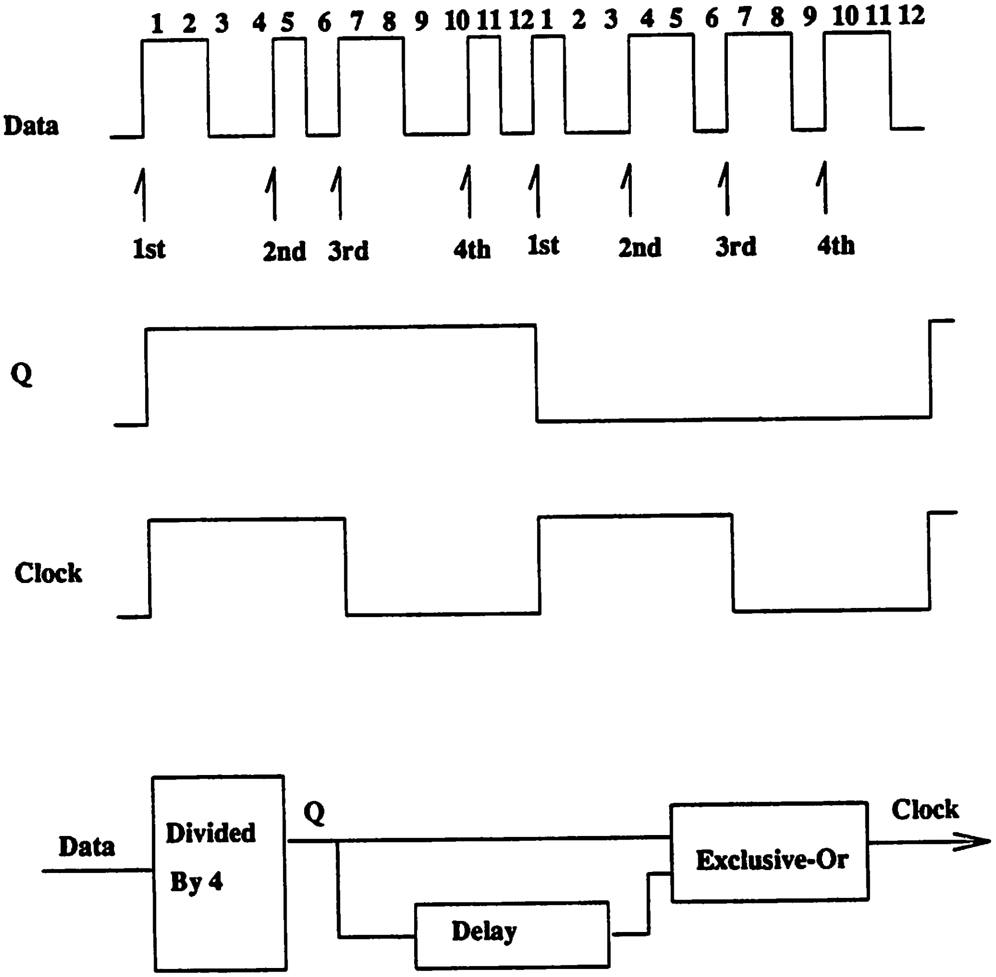


Admissible Codeword 1

4 Transitions from 0 to 1

Admissible Codeword 2

Figure 4. The block timing recovery.



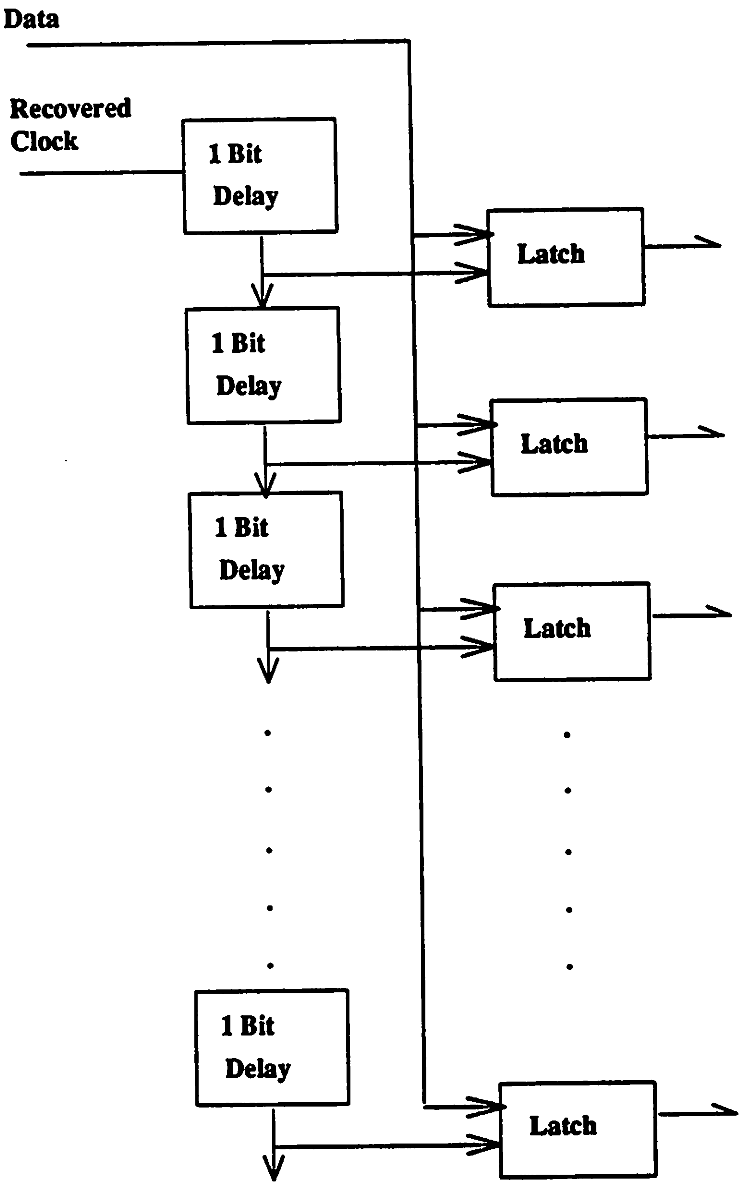
the output of the counter and a delayed version, a square wave clock with a frequency n times lower can be derived. In figure 4, for example, a clock with period n=12 times that of the bit timing is recovered.

The only processing required in this timing recovery approach operating at the data rate is the single divider. An ECL clock divider (prescaler) with input frequency at a few GHz is commercially available [5].

5.2.3. Parallel Data Latching

Given the recovered clock, the remaining process is to latch the received bits in each block. This is achieved by generating a set of delayed versions of the recovered block timing. As shown in figure 5, each delay is approximately one bit period, thereby latching successive (coded) bits of the

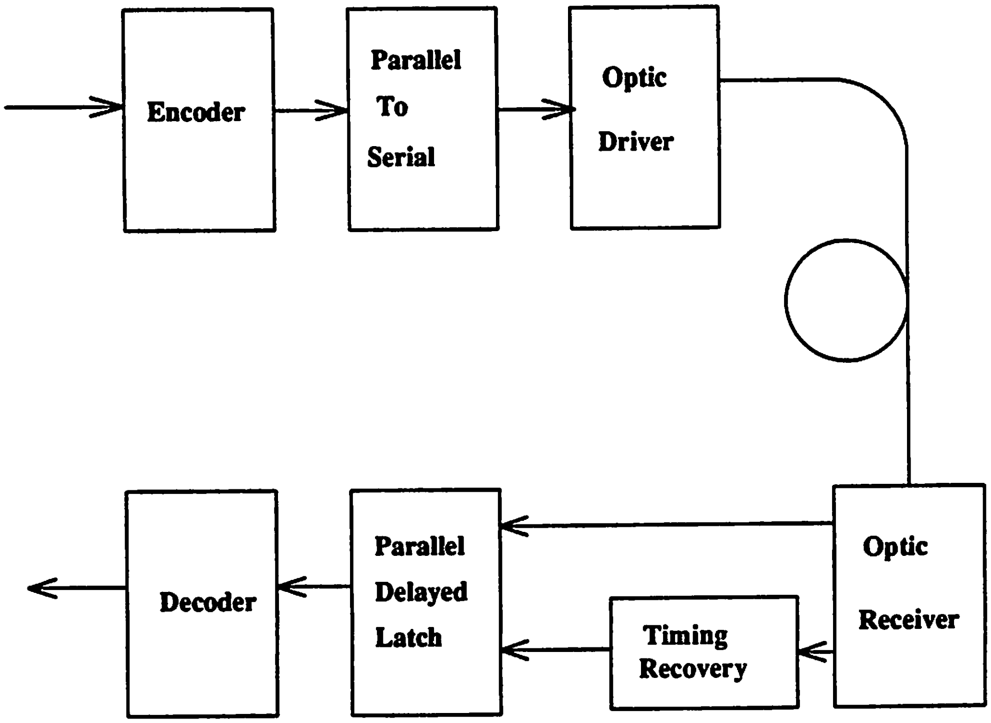
Figure 5. Parallel delay latch lines



block. The serial bit stream is automatically transformed into a parallel format, which is desirable as explained in Figures 1 and 2. In particular, it is consistent with system requirements and reduces the clock rates for subsequent circuitry. The block synchronization function (framing), which is often required, is an inherent part of the timing recovery scheme. Since the delays are only used to latch one block of data, a high degree of accuracy in the actual time delay is fairly relaxed.

A complete view of the system is shown in figure 6. In the transmitter, an encoder generates the fixed transition code. In the receiver, the bit timing recovery and serial to parallel functions are replaced by a single divider and a delay/latch line. A decoder is also included to recover the original data. The code not only simplifies the timing recovery, but also insures that sufficient timing energy is available at all time regardless of the transmitted data. Of importance in LAN networks is the fact that the timing synchronization is almost instantaneous, resulting in a low overhead for synchronization of data packets in asynchronous networks such as CSMA. The presence of line spectral components in

Figure 6. A block view for systems employing the new approach.



the coded data should be of little concern in fiber optics applications where radiation emission is not a problem.

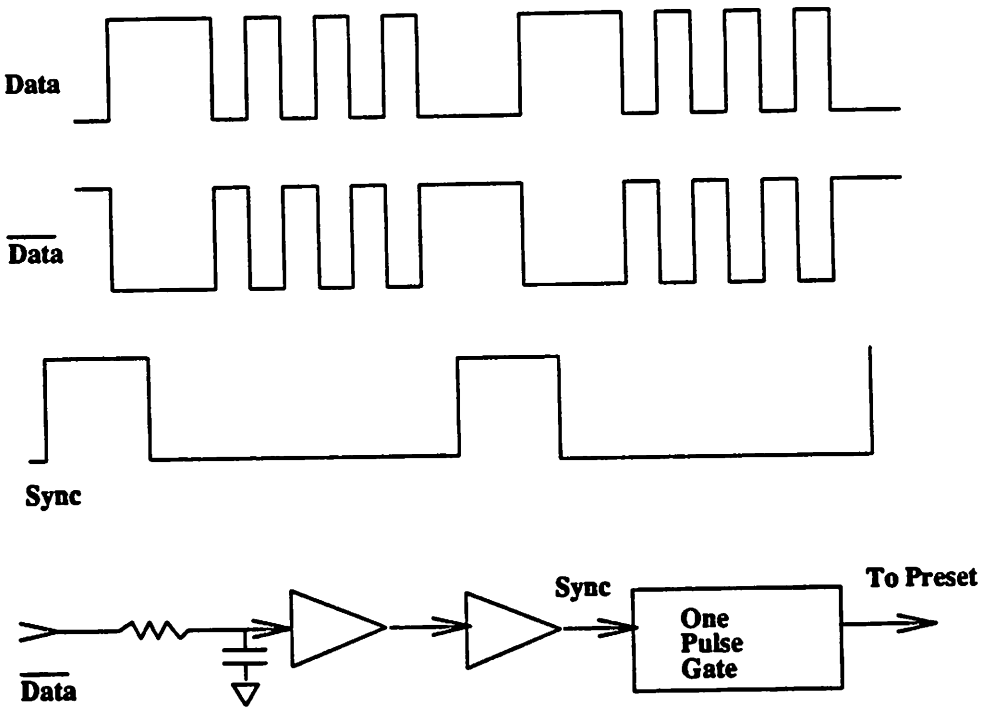
A few technical points remain to be discussed; for example, how to preset the counter at the right timing, the effect of a transition error, the coding efficiency, and the implementation of a precise delay line, All these will be addressed in the next section.

5.3. The Technical Issues

5.3.1. The Block Synchronization and Transition Errors

If the transmission channel gives a good SNR, the received data is self synchronized after the divider circuit. No additional circuitry should be required for the block synchronization. However, in recognition of the possible different characteristics of transceivers used, a simple block synchronization scheme can be added.

Figure 7. The code block synchronization



The key idea is to select a particular block for the preamble sequence so that the receiver can be self-synchronized at the beginning of a transmission period. The desired block contains two parts: one is slowly varying; the other is quickly varying. When it passes through a RC line to a buffer in a lowpass configuration, the output will contain only the slowly varying part and provide the desired preset information for the counter. As illustrated in figure 7, the output of the buffer clears the counter and synchronizes the block.

It is possible (though not probable) to have a lost or extra transition in the bit stream during actual transmission, and as a result to subsequently produce an incorrectly recovered timing. With little noise and at the bit error probability of 10^{-11} to 10^{-12}, the transition error probability in each frame with a size of 1 Kbits to 100 Kbits will be very small and sufficient for most message interleaving transmissions. For continuous bit/byte interleaving transmission like T-carrier systems, a possible solution is to insert a block of preambles periodically to guarantee synchronous blocks. This problem is no different from the normal framing recovery function.

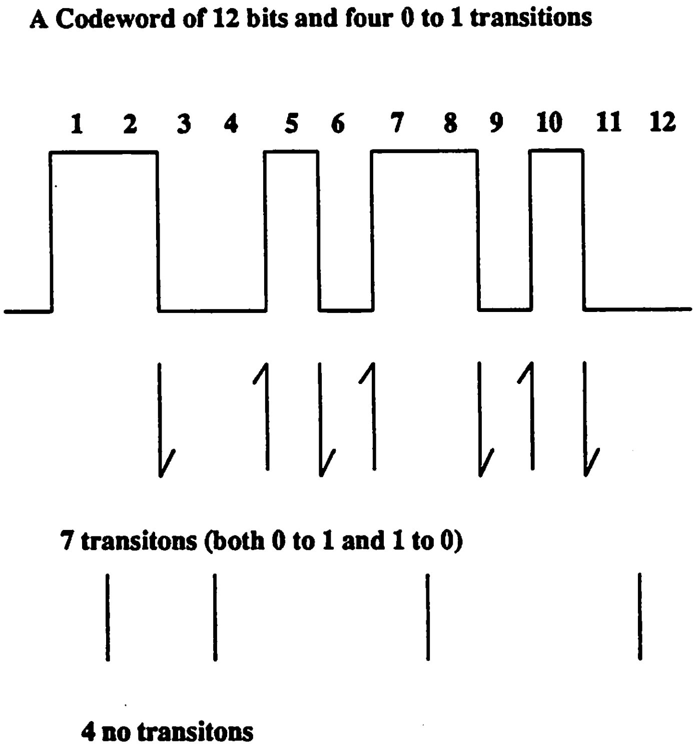
5.3.2. Coding Efficiency and Complexity

Coding efficiency is the percentage of the available bandwidth actually used. Although fiber optics provides inexpensive bandwidth, extra bandwidth does come at some cost in terms of sources, detectors, and preamplifier circuitry, so it is desirable to have an efficiency as high as possible. To calculate the number of possible codewords, we focus on the possible combinations of *transitions*, rather than bits themselves, in each block. As shown in figure 8, there are $2k-1$ possible transitions (both 0 to 1 and 1 to 0) in $n-1$ positions, where the first transition at the beginning of the block is excluded because it's deterministic. Consequently, the number of possible codewords is:

$$S=C(n-1,2k-1) \tag{1}$$

where $C(n,m)\equiv\frac{m!(n-m)!}{n!}$ is the combinatorial function. From this observation, the fixed transition code can also be described as a simple translation from a fixed weight code. The greatest integer value smaller than $\log_2 S$ gives the total possible bit size m of input data. The ratio of m to n is the coding efficiency (or code rate). The efficiency is shown in figure 9 for different values of n. For reasonable

Figure 8. Possible transition locations in a codeword.


A Codeword of 12 bits and four 0 to 1 transitions
1 2 3 4 5 6 7 8 9 10 11 12
7 transitons (both 0 to 1 and 1 to 0)
4 no transitons

block sizes (say n=8 to n=20) the efficiency ranges from about 60% to 80%. The implementation of the encoding and decoding can be done by standard read only memory (ROM) running at the block rate. Larger block sizes result in larger ROMs, but also reduce the speed of the subsequent circuitry. For block sizes of large n, the ROM size of 2^n will be too large to realize. In this case, a combinatorial circuitry to implement the coding and decoding is still possible, and again there will be a tradeoff between simple implementation and high bandwidth efficiency. The Appendix gives a detailed exploration of this combinatorial implementation.

5.3.3. The Precise Delay Line

A precise delay line is required for conversion to block-parallel. While the design of such a device is a problem in integrated circuit design, which we have not addressed in detail, figure 10 shows one approach which shows promise and illustrates the feasibility of meeting the requirements. Two

Figure 9a. The maximum available input block size as a function of code size.

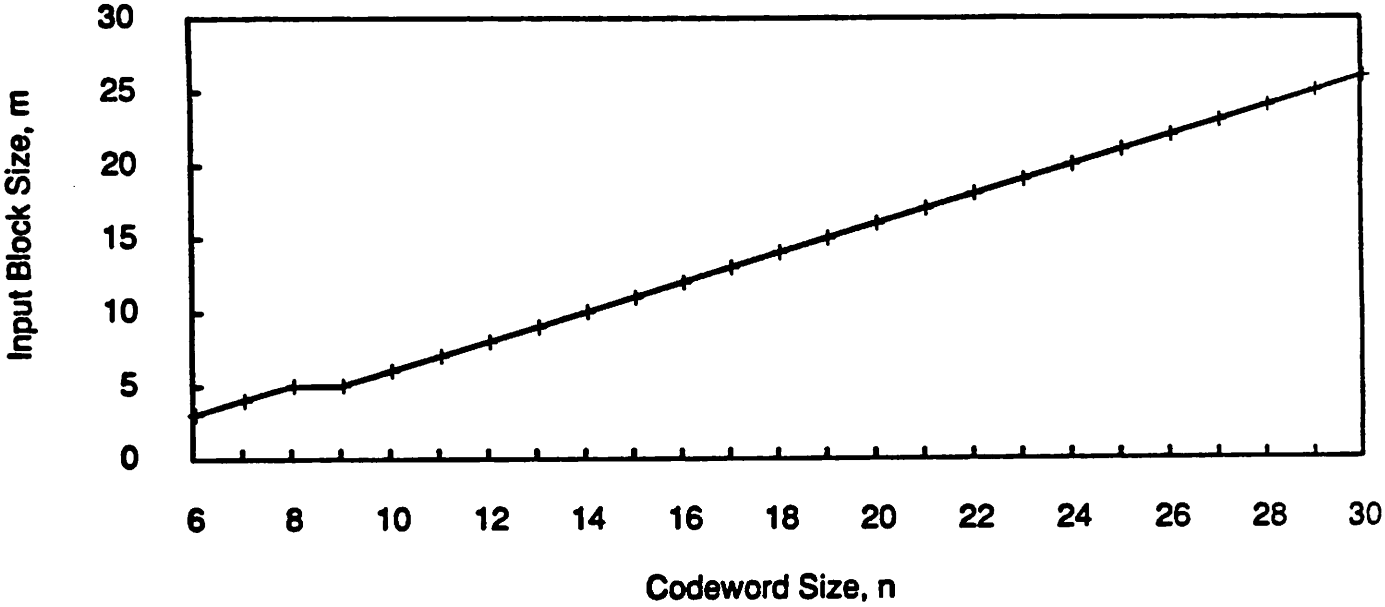
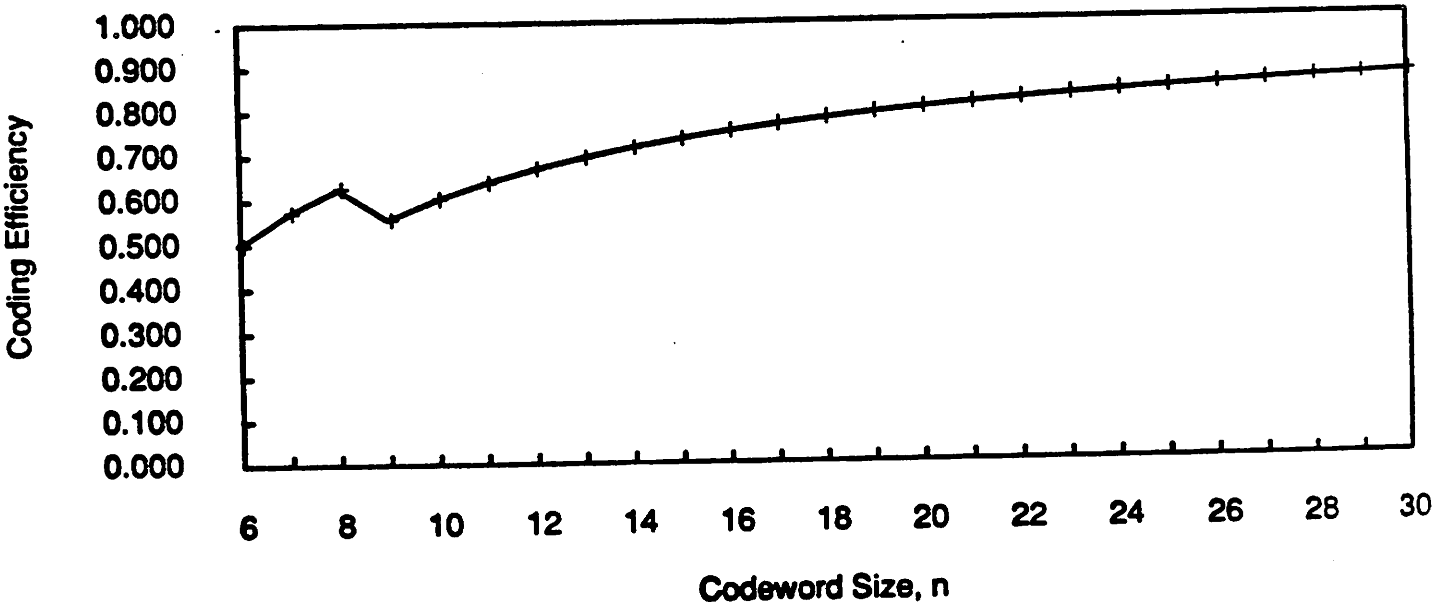


Figure 9b. The coding efficiency as a function of code size.



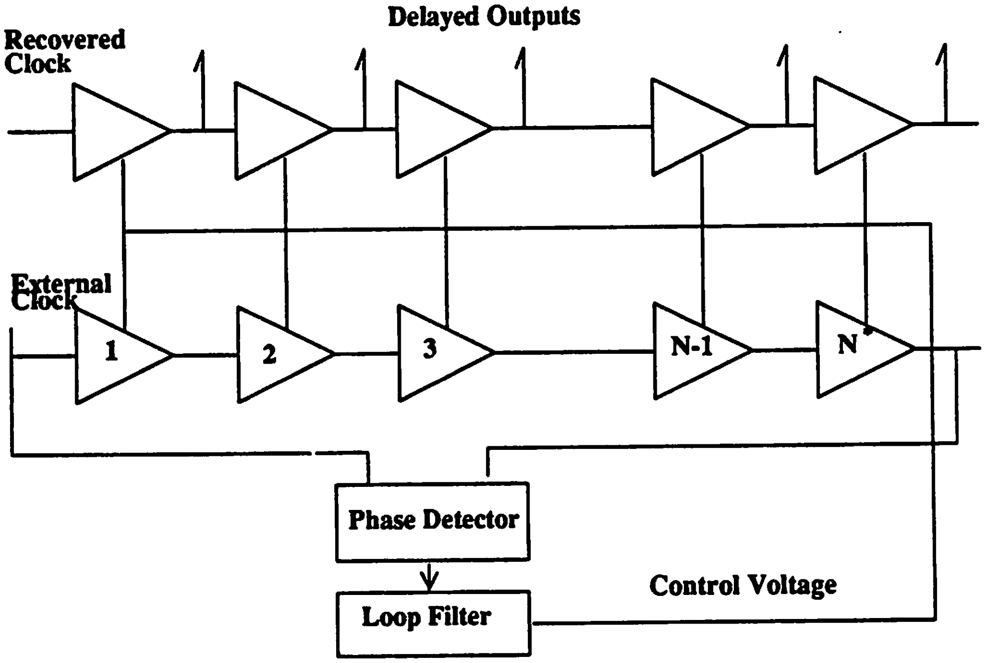
delay lines with identical buffers are used. The delay of each buffer is controlled by a bias current and determined by a common control voltage. By a phase detection of the input and output of one delay line, where the input is from an external source of the same block period (as derived from a crystal oscillator), the control voltage produces the desired delay for another matched delay line. This idea for generating precise delays was similarly proposed and verified in [9].

The delay accuracy is determined by the frequency accuracy of the external clock and the matching of circuitry in two delay lines. The accuracy in matching of the external clock can be very high (e.g. within 0.1%) using a crystal, and the degree of circuit matching can also be high in integrated circuits. Suppose there is a δ% phase inaccuracy at the output of the delay line for the recovered clock, the percentage inaccuracy will be $n\delta$% for the bit clock. Suppose it is required that the latching timing be within +/−ε% of the central data pulse for accurate latching (with no significant degrading in error rate). Then we have

$$n\delta \leq \varepsilon \qquad (2)$$

As a numerical example, with delay phase accuracy of 1% and 25% tolerance in latching, the block size can be as high as n=25.

Figure 10. A precise delay line implementation.



*N is the total bits in one block

5.4. Experimental Examples

For purposes of demonstration, a block code of size $n=12$ and $k=4$ was chosen. This gives $m=8$ data bits per block and a 67% efficiency.

The experimental circuitry is shown in figure 6. Commercial ECL integrated circuits were used, with the exception of a TTL ROM for the fixed transition decoder. For the encoding and decoding, a ROM of 256 by 10 bits is sufficient (2.5K bits) for the encoder, and the decoder is a 1024 by 8 ROM. The block synchronization design follows the approach suggested in the previous section, and the delay line was implemented by a series RC delay buffer, where the delays were set manually. The optic transceivers used were the AT&T ODL-50 and ODL-200, connected with multimode fibers.

Data rates of 50 Mb/s and 200 Mb/s were tested. Figure 11 shows the transmit and receive data block. Only the preamble sequence was transmitted, and the recovered block timing was self-synchronized. The delayed block timing of 205 nsec which latched the 11th bit is also shown. Figure 12 shows two data blocks transmitted alternatively, and the block timing was still maintained correctly.

5.5. Conclusion

The fixed transition coding method provides simple timing recovery by having the transmitter combine the data and timing information. We believe that the timing recovery method presented in this chapter provides an attractive solution to the timing recovery problem in future optic networks.

Figure 11a. 50 Mb/s. (a) Transmitting data (the preamble: 111010101000); (b) Received data, via AT&T ODL-50; (c) Recovered Timing (240 nsec/block).

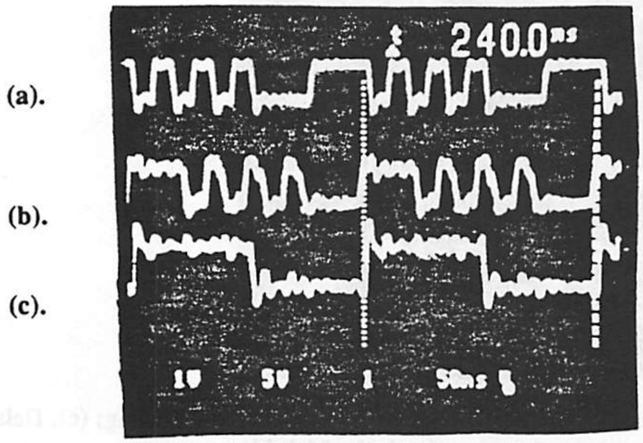


Figure 11b. (a) Received data; (b) Recovered Timing; (c). Delayed clock to latch the 11th bit.

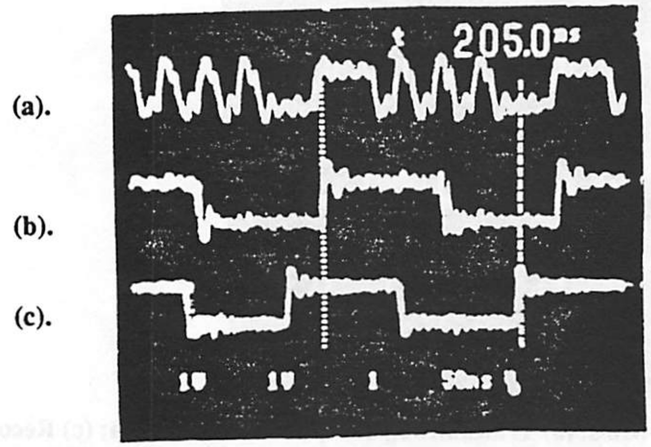


Figure 11c. 200 Mb/s. (a) Transmitting data (the preamble); (b) Received data, via AT&T ODL-200; (c) Recovered Timing (60 nsec/block); (d) Delayed Clock.

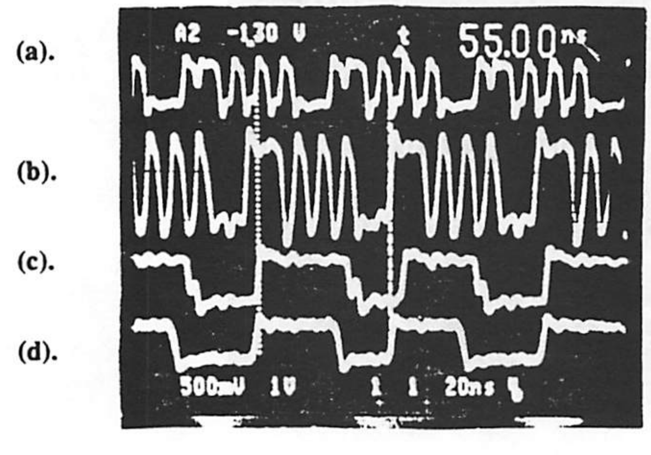


Figure 12a. 50 Mb/s. (a) Transmitting data of two different blocks; (b) Received data; (c) Recovered Timing (240 nsec/block).

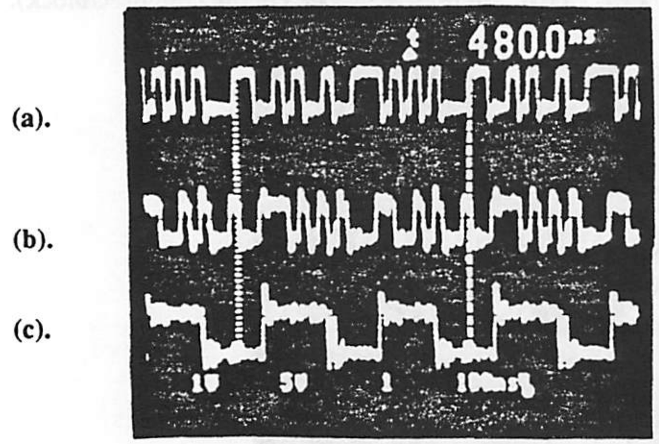


Figure 12b. 50 Mb/s. (a) Received data; (b) Recovered Timing; (c). Delayed clock to latch the 11th bit.

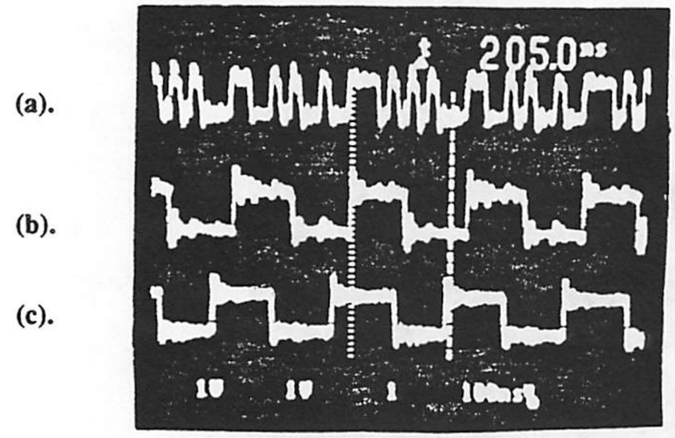
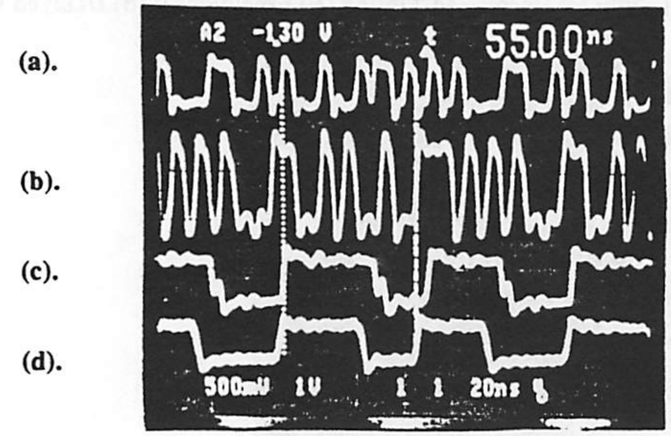


Figure 12c. 200 Mb/s. (a) Transmitting data; (b) Received data; (c) Recovered clock (60 nsec/block); (d). Delayed clock.



Appendix. Algorithms for Encoding and Decoding

A codec design for the mapping between a fixed transition codeword and a binary data word is considered in this appendix. First, the equivalence between the fixed weighted and transition codes is explained, and the circuit implementation is simple and straightforward. As shown in figure A.1, fixed transition codes and fixed weighted codes can be converted simply with just exclusive-or gates. In the following, without loss of generality, a coding algorithm and design of the fixed weighted code of (11,4) is illustrated, where each block has 11 bits, with 4 bits of 1's. The total possible number of codewords is 330, allowing binary source data words of 8 bits.

The Mapping. First, the codeword is listed from the smallest binary value (00000001111) to the largest (11110000000). The mapping is defined by assigning each codeword an 8 bit binary number whose value is equal to the sequence number of the codeword in the list. In other words, 00000001111 is mapped to 00000000 (0), 00000010111 to 00000001 (1), and so on (Table 1).

Figure A.1. Conversion between Fixed Transition Code and Fixed Weighted Code.

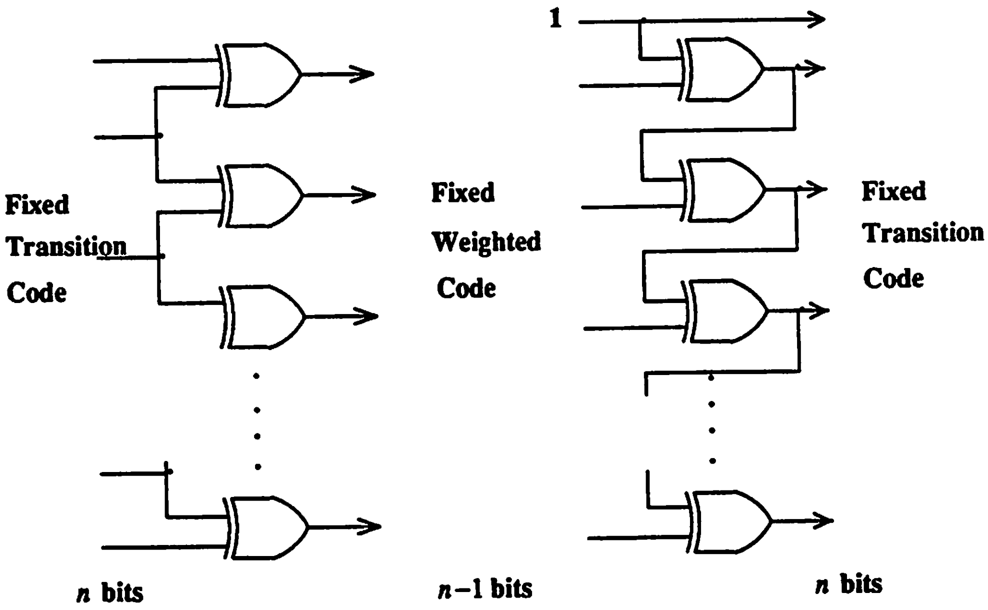


Table 1, The mapping between data and the fixed weighted code (11,4).

decimal data	binary data	codeword
0	00000000	00000001111
1	00000001	00000010111
2	00000010	00000011011
3	00000011	00000011101
4	00000100	00000011110
5	00000101	00000100111
	. . .	
254	11111110	10010011000
255	11111111	10010100001

The Algorithm. To get an arithmetic algorithm to implement the mapping, some properties are observed:

(1) For each "1" in the codeword, depending on its positions in the whole block and the sequence relationship with the three other "1"s, it has a corresponding "**base value**". In other words, if a "1" of a codeword is in a position i counting from left and has $j-1$ "1"s on its left hand side, its base value is defined b_{ij}. For the (11,4) fixed weighted code, the corresponding base values for each "1" in the codeword is shown in table 2.

(2) The summation of the four base values in binary is the mapped data.

Table 2, Base value b_{ij} for the fixed weighted code (11,4).

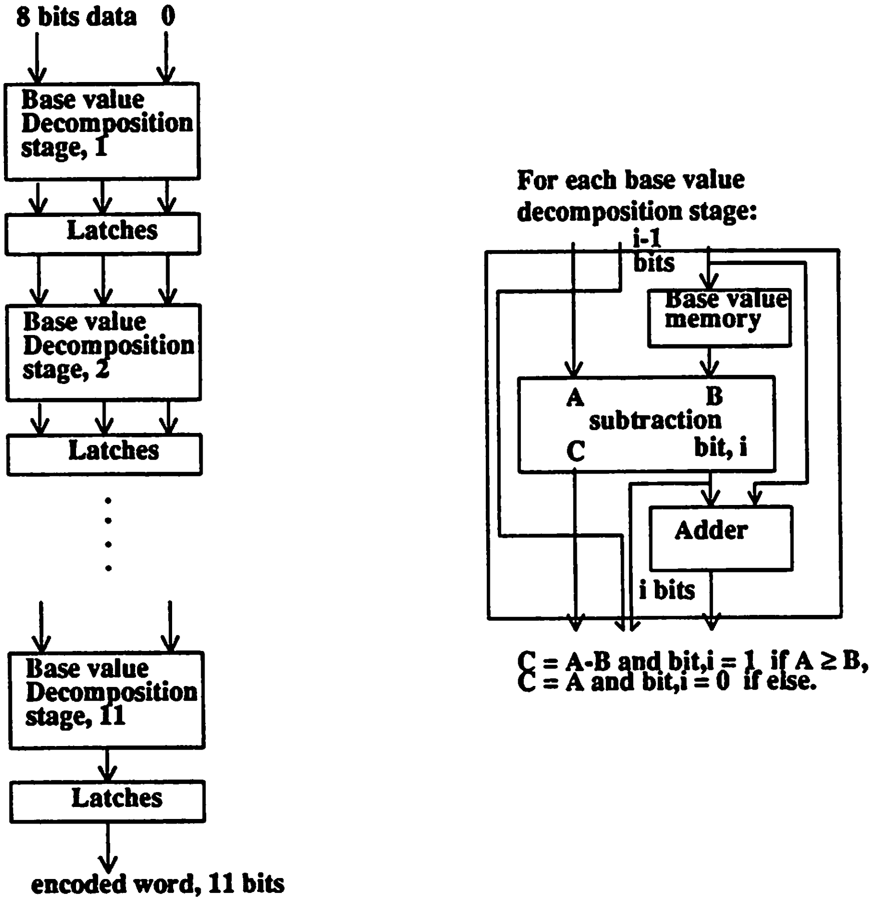
j	i										
	1	2	3	4	5	6	7	8	9	10	11
1	210	126	70	35	15	5	1	0	-	-	-
2	-	84	56	35	20	10	4	1	0	-	-
3	-	-	28	21	15	10	6	3	1	0	-
4	-	-	-	7	6	5	4	3	2	1	0

i: the position of each bit in the codeword from the left hand side,
j: the position for "1" in the codeword among all "1"s, from left.

As an example, a codeword (00101100100) is mapped to 70+20+15+5=110= $(01101110)_2$, where b_{31} is 70, b_{52} is 20, and so on. On the other hand, if a binary number is given, for example 245=$(11110101)_2$, it can be expressed as a sum of 210+35+0+0, the codeword it maps is (10010000011). Notice that the mapping is unique.

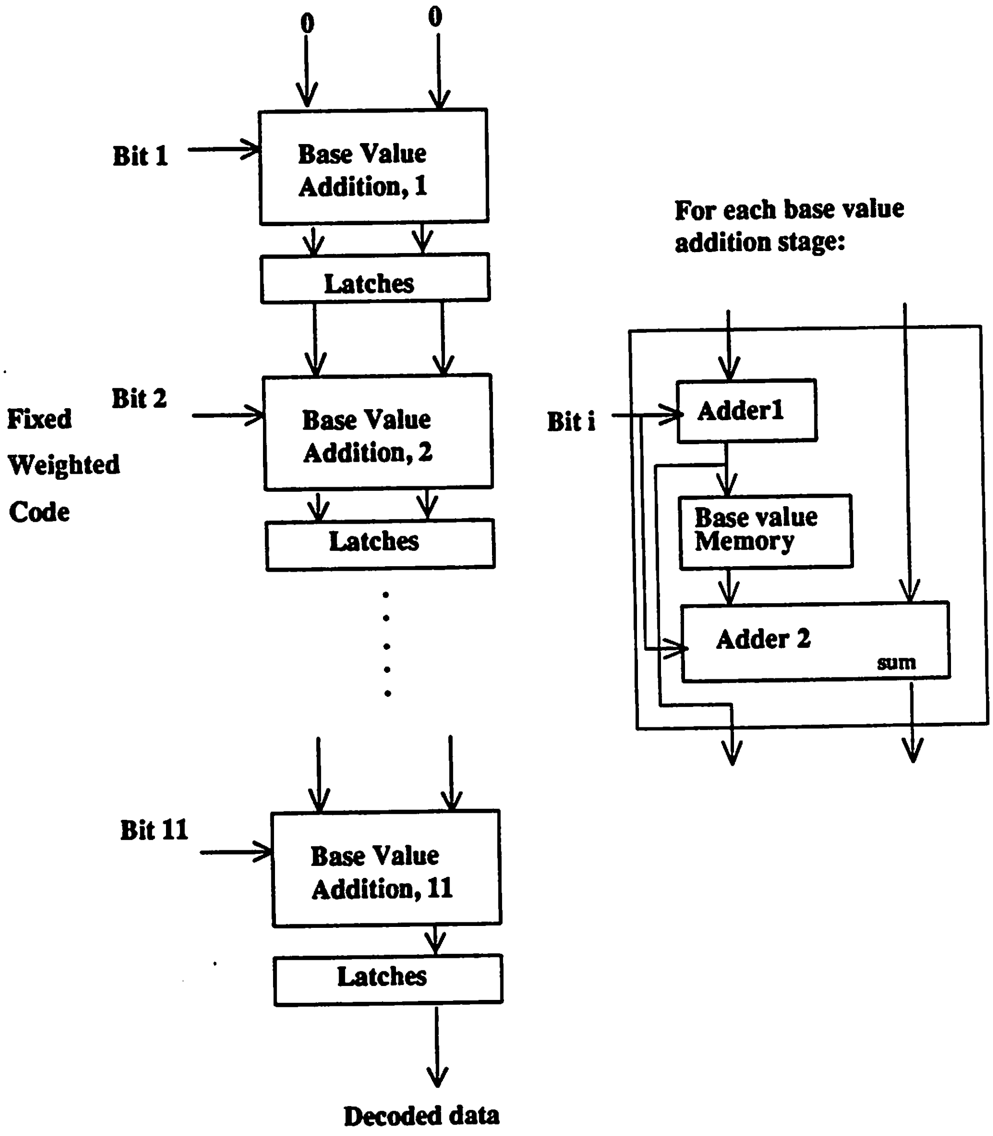
The Implementation. It consists of two parts: encoding and decoding. The encoder needs to decompose the binary input into a sum of four base values, and outputs "1" at the corresponding bits and "0" elsewhere. The logic diagram is shown in figure A.2. In the first decomposition stage, the input is the 8 bit data. If the data is greater than b_{11}, it is subtracted by b_{11} and the output bit_1 is set to 1. Otherwise there is no subtraction, the data is passed to the next stage, and the output bit_1 is 0. This

Figure A.2. Encoder, where data is decomposed into a summation of base values.


8 bits data
0
Base value Decomposition stage, 1
Latches
Base value Decomposition stage, 2
Latches
Base value Decomposition stage, 11
Latches
encoded word, 11 bits
For each base value decomposition stage:
i-1 bits
Base value memory
A
B
subtraction
C
bit, i
Adder
i bits
C = A-B and bit,i = 1 if A ≥ B,
C = A and bit,i = 0 if else.

same process is carried through in the following stages, and the adder, whose output is used to address the base value among four rows for each bit in table 2, is incremented by 1 if the stage has output 1. It is noted that each base value memory has only four words for code (11,4). As shown in the figure, a pipeline structure can also be formed by inserting latches between each stage so the high speed limit can be avoided.

Figure A.3. Decoder, where the correct base values corresponds to each "1" in the input codeword is accumulated and the final summation is the desired data.



The decoder does the inverse operation by fetching correct base values and summing them, figure A.3. The first adder in each stage is used to address the correct base values among four rows in table 2 for each bit. The second adder accumulates the correct base value in each stage if the input bit is 1; otherwise, the adder does nothing. As noted, latches can also be added for the same reason above. For a whole system picture, the encoding and decoding is shown in figure A.4.

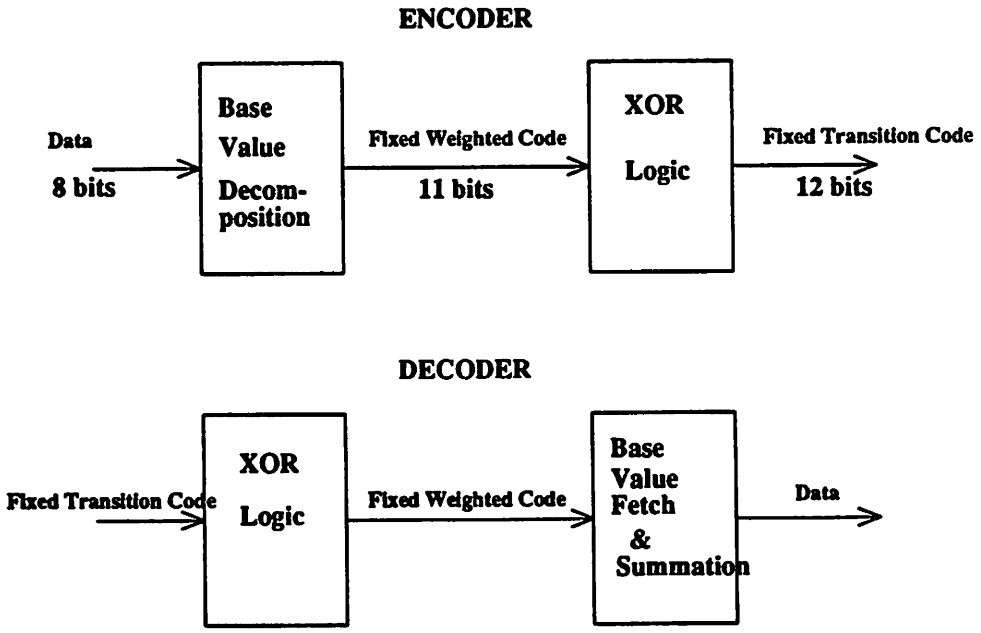
REFERENCE:

[1] Ming-Kang Liu, David G. Messerschmitt, and David A. Hodges, *Time Slot Switching and Fiber Optics for Integrated Service of PBX/LAN*, submitted to Trans. on Communications

[2] Robert M. Metcalfe et. al. *Ethernet: Distributed Packet Switching for Local Computer Networks*, Comm. of ACM, July 1976, pp.395-404.

[3] B. E. Kaiser and E. Strange, *Digital Telephony and Network Integration*, Van Nostrand

Figure A.4. A systematic view of the encoder and decoder.



Reinhold Pressed, 1985.

[4] Kamilo Feher, *Digital Communications, satellite/earth station engineering,* **chapter 8, written by S. J. Campanella and D. Schaefer, Prentice-Hall press, 1981.**

[5] Motorola ECL Data Book, Chapter 7.

[6] Trudy E. Bell *Technology '85, Communications,* **IEEE Spectrum, January 1985, pp. 56-59.**

[7] Y. Takahashi, et. al., *Integrated Optical Repeaters Using Long Wavelength LED/LD for High Speed Computer Networks,* **pp. 1355-1358, Globecom 1984.**

[8] Kazuo Hagimoto, et. al., *Very High Speed Intra-Office Optical Link Experiments for 0.4- and 1.6-Gb/s Trunk Lines,* **IEEE J. of Lightwave Technology, Oct. 1984, pp668-675.**

[9] Allen G. Bell and Gaetano Borriello, *A Single Chip NMOS Ethernet Controller* **ISSCC 83, pp.70-71.**

[10] H. J. Shin, *Private communication,* **University of California, Berkeley.**

CHAPTER 6

The Cascading Effect of Circuit Switches

6.1. Introduction

Circuit switches were suggested in chapter 2 to provide a higher network throughput for a given technology. To minimize the low speed constraints from electronics in optical networks, no timing recovery in the switch is one of the primary features proposed in TSS. That is, the signal passing intermediate switching nodes is converted from optical pulses to electrical pulses, amplified, switched, and transformed to optical pulses again. In the process, no retiming is used.

As pointed out in chapter 5, high speed timing recovery beyond 100 Mb/s in general is not easy to accomplish. Therefore, avoiding retiming process in the intermediate switches provides a higher speed capability and lower cost implementation. However, the associated disadvantage is a larger timing jitter which may accumulate through many switch stages, especially if the circuit switch has larger crosstalk between adjacent channels. As a result, this chapter provides a preliminary study and design for this problem. At the time of this thesis is written, the high speed circuit switch chip [8] is still not available; therefore, a logic circuit is used instead to simulate the switching chip. Further work is discussed at the end of this chapter.

6.2. Timing Jitter Model

For digital transmission, a timing signal is recovered at the receiver to regenerate data bit streams. Because noise and dispersion are inherent in the channel and timing recovery is imperfect in practical realizations (e.g., mistune in the bandpass resonator), timing jitter is always associated with the recovered timing signal. This timing jitter degrades the receiver performance.

In this chapter, the experiment part was primarily carried out by Janet M. Cooper, [7].

Generally speaking, the magnitude of the timing jitter increases when transmission distance increases. To overcome this problem, repeaters with retiming are used between the source and destination. In each repeater, there is a local timing derived from the input data stream. Derived by either a phase lock loop (PLL) or a narrow band filter, this timing is insensitive to the incoming jitter associated with the input data. The jitter after retiming can thereby be greatly reduced.

In telephone systems, a long distance transmission line may consist of many repeaters for the above purpose. The jitter effect after cascading these repeaters becomes very significant at the final receiver. Consequently, much work has been done on analytic models and experimental tests [1-6]. In general, the jitter can be classified into random and systematic jitter. The random jitter comes from random noise in the channel [1], while the systematic jitter comes from the data pattern dependence in the timing recovery [6]. Because of this dependence, the systematic timing jitter will accumulate through the repeater chain and is a more serious problem.

In TSS, timing is not recovered in the switches until in the final receiver, and the timing jitter introduced in each stage is primarily from asynchronous channels and the switching circuit itself (e.g. thermal noise). Therefore, the random timing jitter is more significant than the systematic jitter. If each stage introduces a jitter with independent and identical statistics (IID), and the jitter in each stage is Δt, the final jitter accumulated after n stages will be:

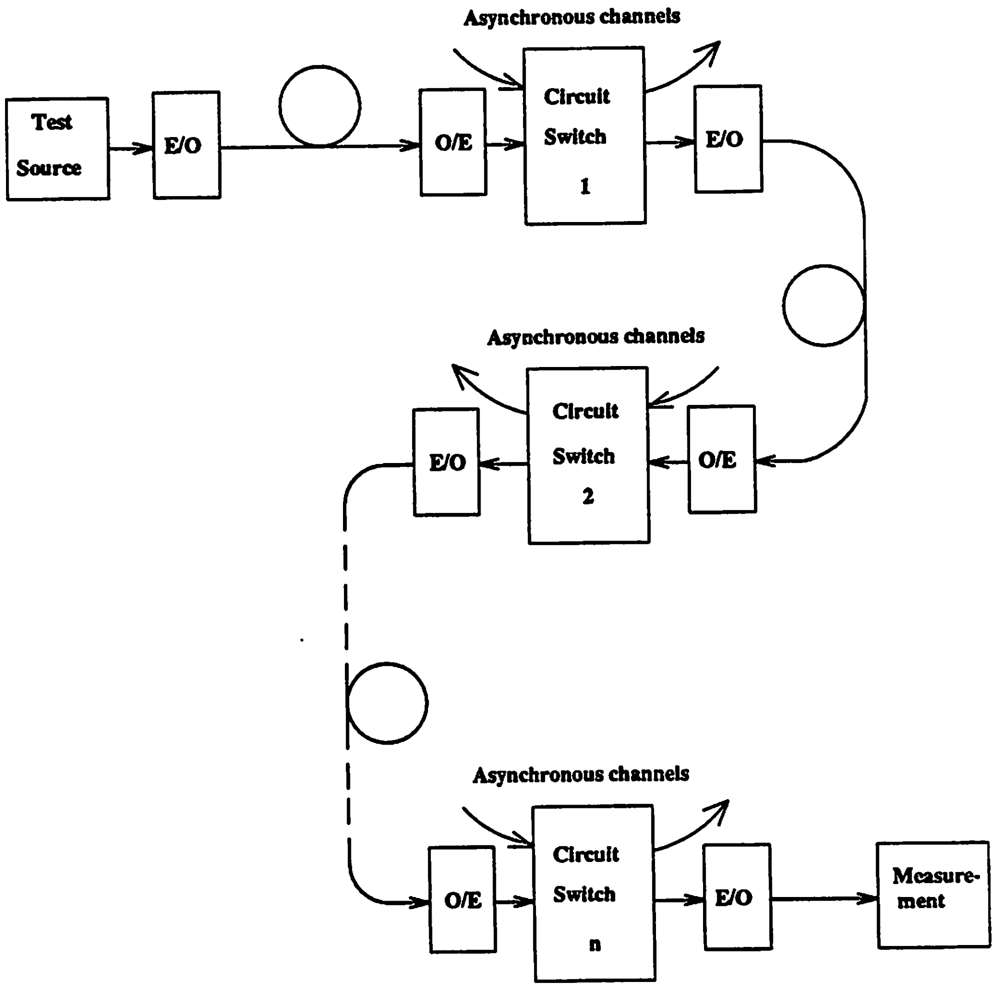
$$\Delta t_n^2 = n\,\Delta t^2 \qquad (1)$$

So the magnitude of the jitter will be proportional to the square root of the number of stages in cascade. In LAN applications, both the jitter size Δt in each stage and the number of stages are small; therefore, timing recovery is not necessary. For high speed communication, this is very attractive because of low implementation cost and little degraded performance.

6.3. Experiment Setup

The purpose of the experiment is to further justify the feasibility of no timing recovery through cascaded circuit switches. A logical view for the testing system is shown in figure 1. As noted, it consists mainly of optical transceiver links. In practice, available optical fibers and transceivers on hand

Figure 1. A logical view for the jitter testing system.

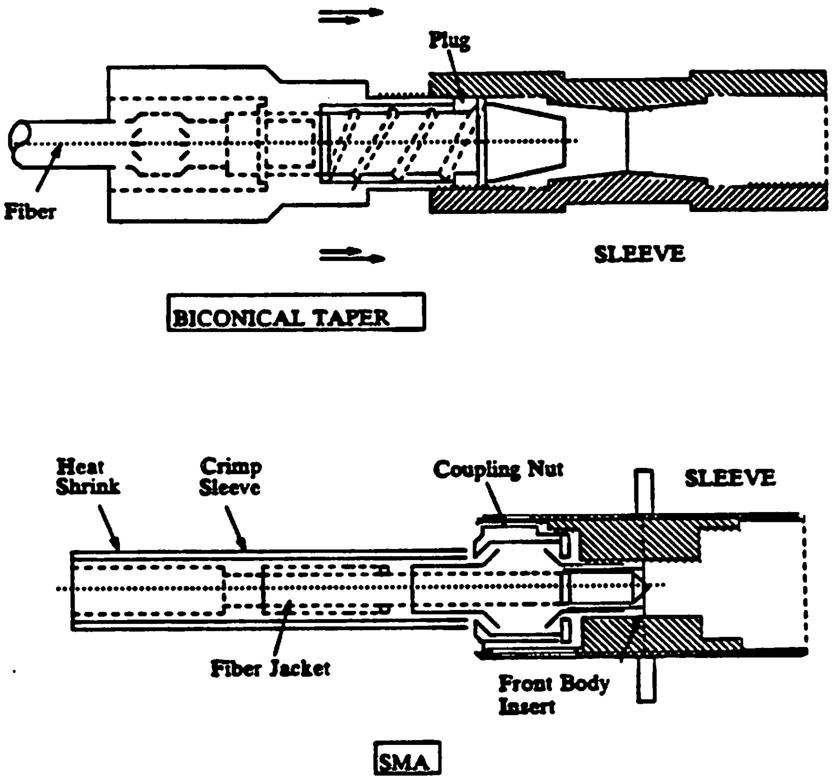


are very limited. Consequently, a feedback loop structure, where only one optical link is needed, is used to replace the structure shown in figure 1. The characteristics of the optic links used and the feedback loop design are described in the remaining section.

6.3.1. Optical Link Components

In the experiment, the optical components, which consist of optical fibers, fiber connectors, optical transceivers, and fiber connection tools are all from AT&T Bell Labs. The fibers used are all

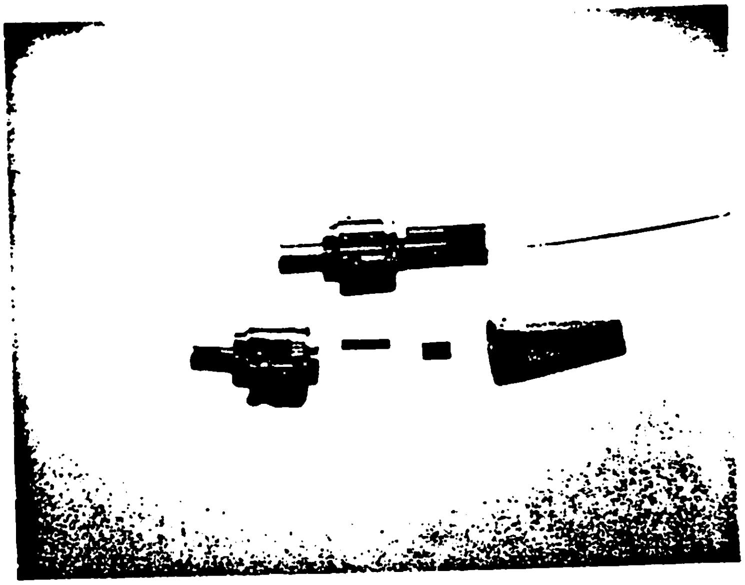
Figure 2. Illustrations of biconical and SMA connectors.



62.5/125 μm (core/cladding) multimode. Their distance ranges from 400 to 700 meters (by experimental testing). Multimode fibers offer large mechanical tolerance in fiber splicing and will be sufficient for this preliminary testing.

The two most popular connector types are shown in figure 2. These connectors are low loss and available in multiple-mode and single-mode versions [9]. The biconical connector, invented and standardized by AT&T, is larger than SMA due to its internal spring and the biconical design. The connector used in the experiment, however, is called the *ST™* Series Lightguide Connectors, a combination of the above two, figure 3. Depending on preparation, these connectors can have a loss as low as 0.2 db [7].

Figure 3. Photographs of the *ST™* connector used in the experiment.



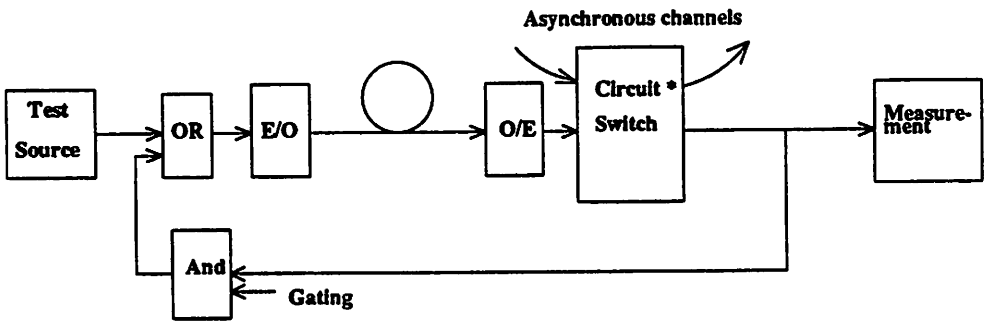
The optical transceiver pairs used are ODL-50 and ODL-200 Lightwave Series. ODL-50 is designed for data rates from 1 to 50 Mb/s (NRZ) at distances up to 3 Km. The transmitter has a LED operated at 0.85 μm wavelength, and the receiver has a PIN photodiode for detection. ODL-200 is designed for data rates from 40 to 220 Mb/s (NRZ) at distances also up to 3 Km. Its LED is operated at 1.3 μm and it also uses PIN diode for detection. Both ODL-50 and ODL-200 are compatible with 62.5/125 μm fiber and are available in standard 16-pin DIP packages.

The fibers were originally obtained from AT&T separately from the connectors. Tools are needed to join them. These wiring tools are included in the AT&T 1032A Tool Kit for multimode fibers and *ST™* connectors wiring. Detailed connection procedures can be seen in [7,10].

6.3.2. Feedback Loop Setup

A practical system for jitter testing is shown in figure 4. The primary motivation to have such a structure is to use only one optical link to study the switch cascading effect. The basic idea is to feed

Figure 4. A practical testing system in feedback loop for jitter test.



* : Simulated by combinatorial logic circuits.

the received pulse back to the transmitter, so to cascade the circuit switches is equivalent to passing the pulse as many times as the number of switches in cascade. If the optical channel has large bandwidth so that the intersymbol interference is insignificant, this approach provides very economic and simple implementation.

If no special care is taken, the pulse may recirculate infinitely through the same link. To control the number of recirculations, which is also the number of switches cascaded, a gating signal is used to kill the feedback pulses after recirculating a certain number of times, figure 5.

In a practical implementation, the source signal is a periodic pulse train with a small duty cycle, and the gating signal derived from it is a wider inverted pulse with almost the same phase, figure 6. As a result, to kill the n th pulse, a relation between the propagation delay T_{pg} and the pulse period T_p should be satisfied:

$$n \cdot (T_{pg} \; mod \; T_p) = T_p \qquad (2)$$

Since a deterministic pulse source is used, only a "half eye" diagram can be observed for the jitter accumulation effect. As shown in figure 4, a combinatorial logic circuit is inserted in the feedback loop to simulate the circuit switch. Asynchronous signals can be excited to investigate the "crosstalk" effect to the timing jitter. When the real high speed circuit switch becomes available, a more realistic jitter study can be carried out.

6.4. Test Results

The results for jitter accumulation based on the description above are summarized in this section. Although they are very preliminary and qualitative, they suggest that not including retiming in intermediate switching nodes is very promising. Since the ODL-50 and ODL-200 have different frequency ranges, they were tested separately.

Figure 5. Feedback timing for the system in figure 4.

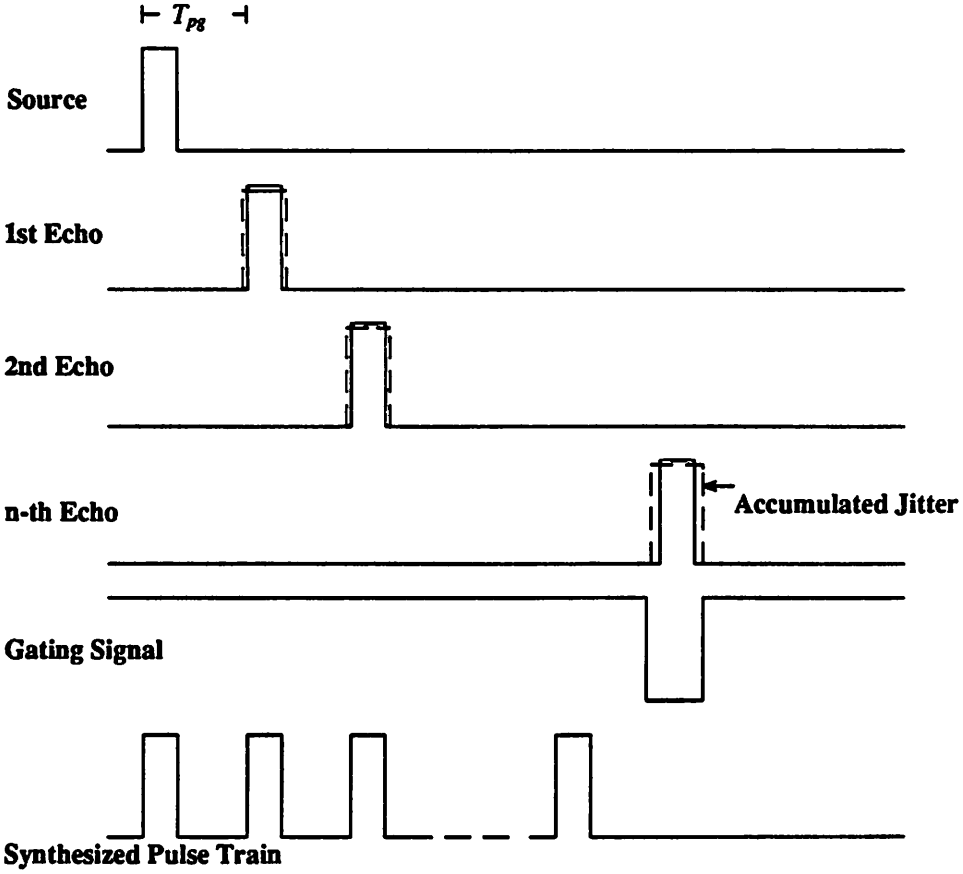


Figure 6. Illustration of timing relationship in Eq. (1).

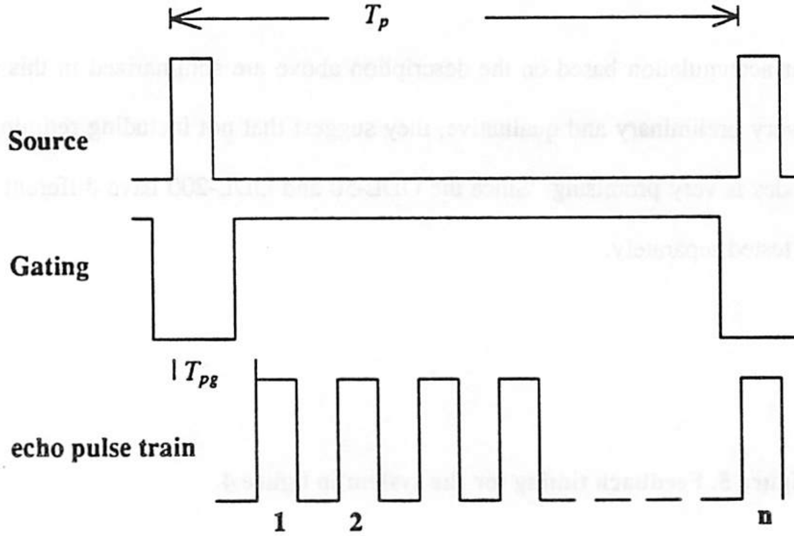
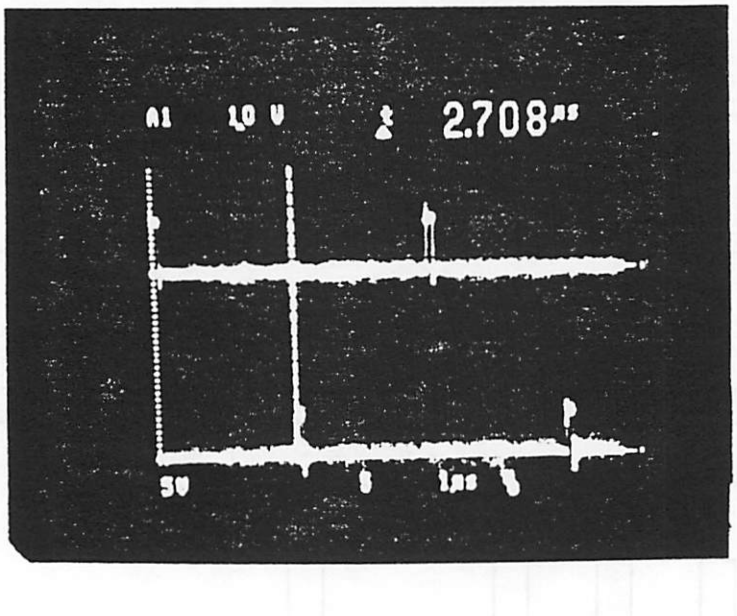


Figure 7. The propagation delay measurement.



6.4.1. ODL-50

For ODL-50, input pulse widths from 20 nsec (50 Mb/s NRZ) to 1 μsec (1 Mb/s NRZ) are tested. Figure 7 shows a fiber of propagation delay of 2.7 μsec is used. Figure 8 shows the transmitted pulse train plus the recirculated pulses for up to 1, 2, 5, 35, 55 passes through the loop. All the pulses are of width 150 nsec. All the results in figure 8 have no asynchronous noise excited.

As shown in the figure, the timing jitter effect is not significant even for a large number of recirculations. The results for narrower width (20 nsec) and with asynchronous noise excited are similar and show very little degradation. The photo results are not shown here because they are indistinguishable from camera pictures.

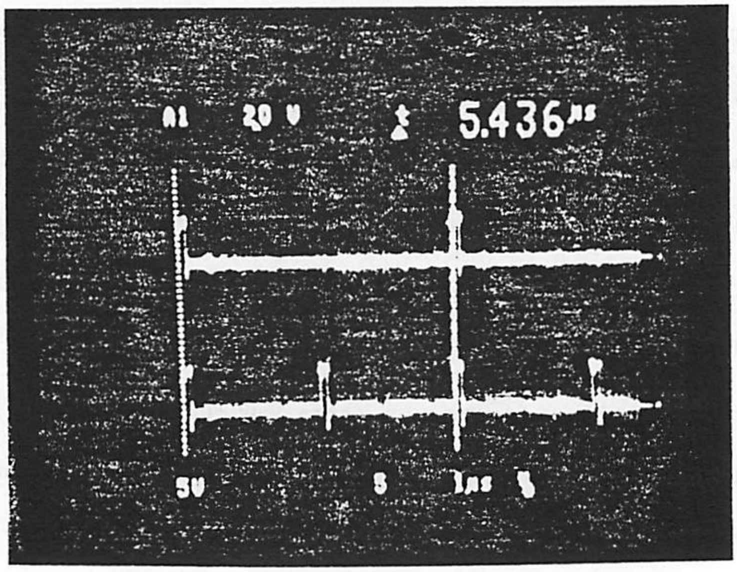
Different fiber lengths (6.06 μsec with 1 extra connector between fibers and 8.86 μsec with 2 extra connectors) for the same testing are also investigated. Basically they show the same results as mentioned above.

6.4.2. ODL-200

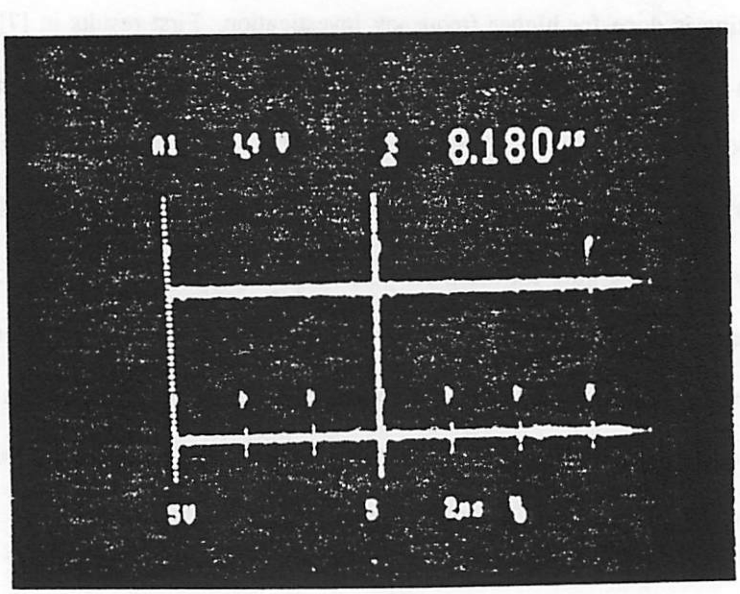
ODL-200 operates at a higher frequency range and is ECL 100K series compatible; therefore, separate testing is done for higher frequency investigation. First results in [7] showed unsuccessful transmission except for 50% duty cycle pulse trains. This makes the original feedback loop design in figure 4 impossible. With an additional testing circuit board supplied by AT&T Bell Labs, successful transmissions with different duty cycle percentages are observed, as shown in figures 9,10.

To test the timing jitter for narrower pulse widths (i.e., higher transmission rate up to 200 Mb/s NRZ), the same circuit design in figure 4 can be used. This is feasible because of successful short pulse transmission verified in figures 9,10. Since the high speed circuit switch is not available and it has its own timing jitter and crosstalk characteristics, the investigation of the cascading effect for high rate at 200 Mb/s has been delayed until the above situation is resolved.

Figure 8. Input and recirculated pulses through ODL-50 and 62.5/125 fibers.



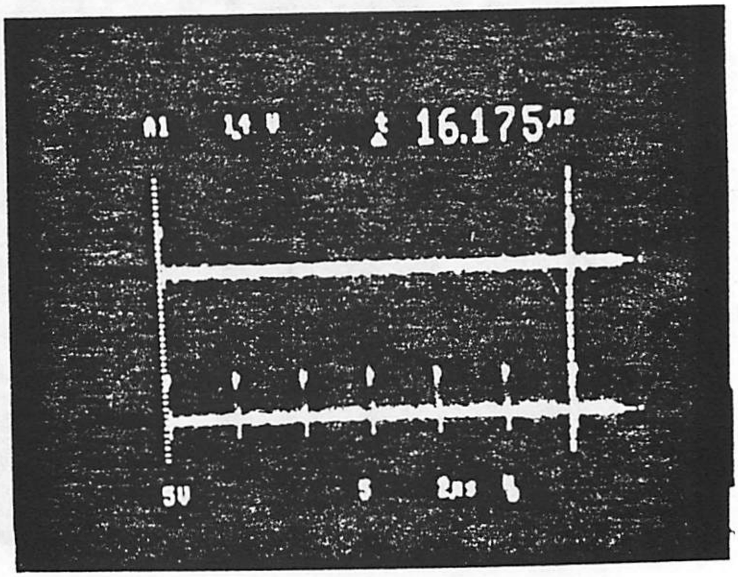
(1). $T_p=2T_{pg}$



(2). $T_p=3T_{pg}$

Figure 8. continued.

(3). $T_p=6T_{pg}$



(4). $T_p=36T_{pg}$

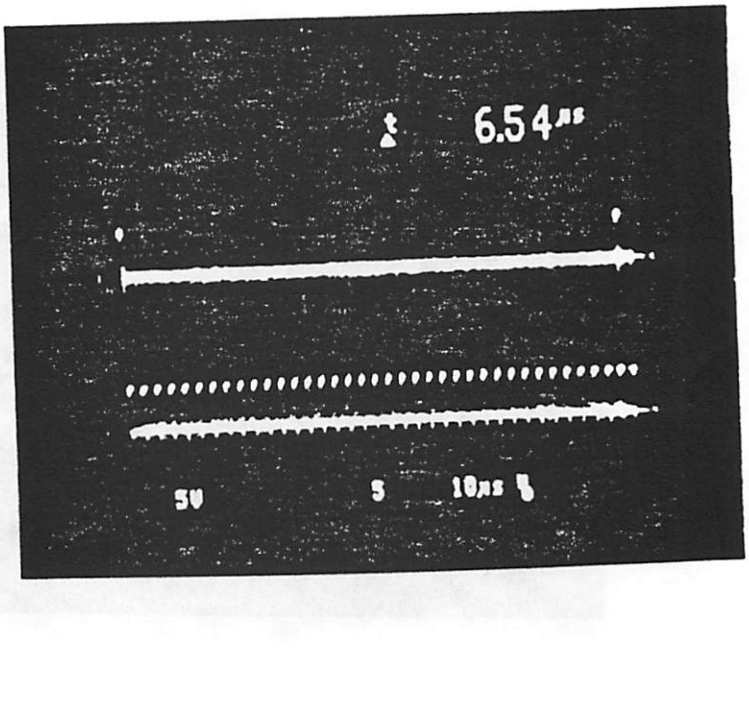


Figure 8. continued.

(5). T_p=56T_{pg}

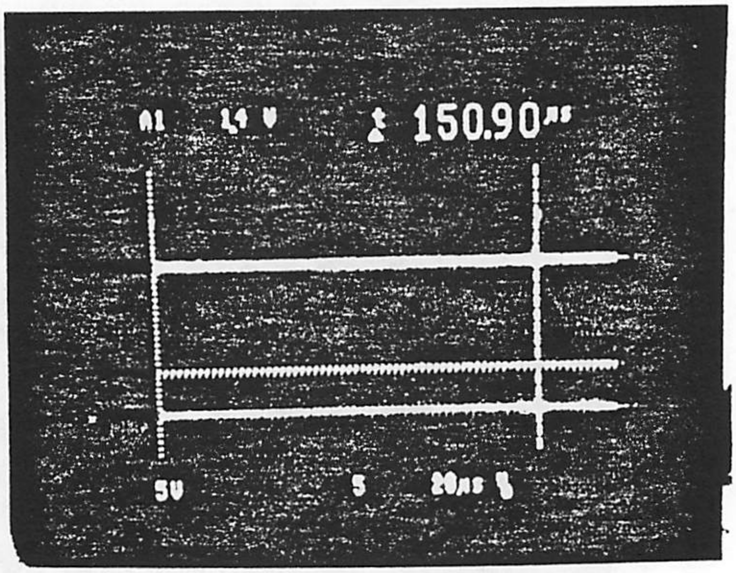


Figure 9. The tested clock transmission via ODL-200.

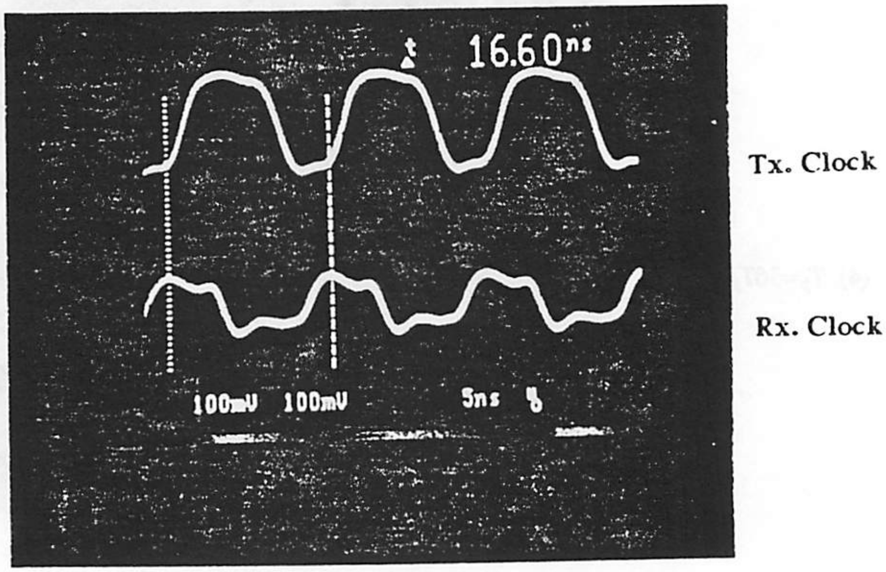
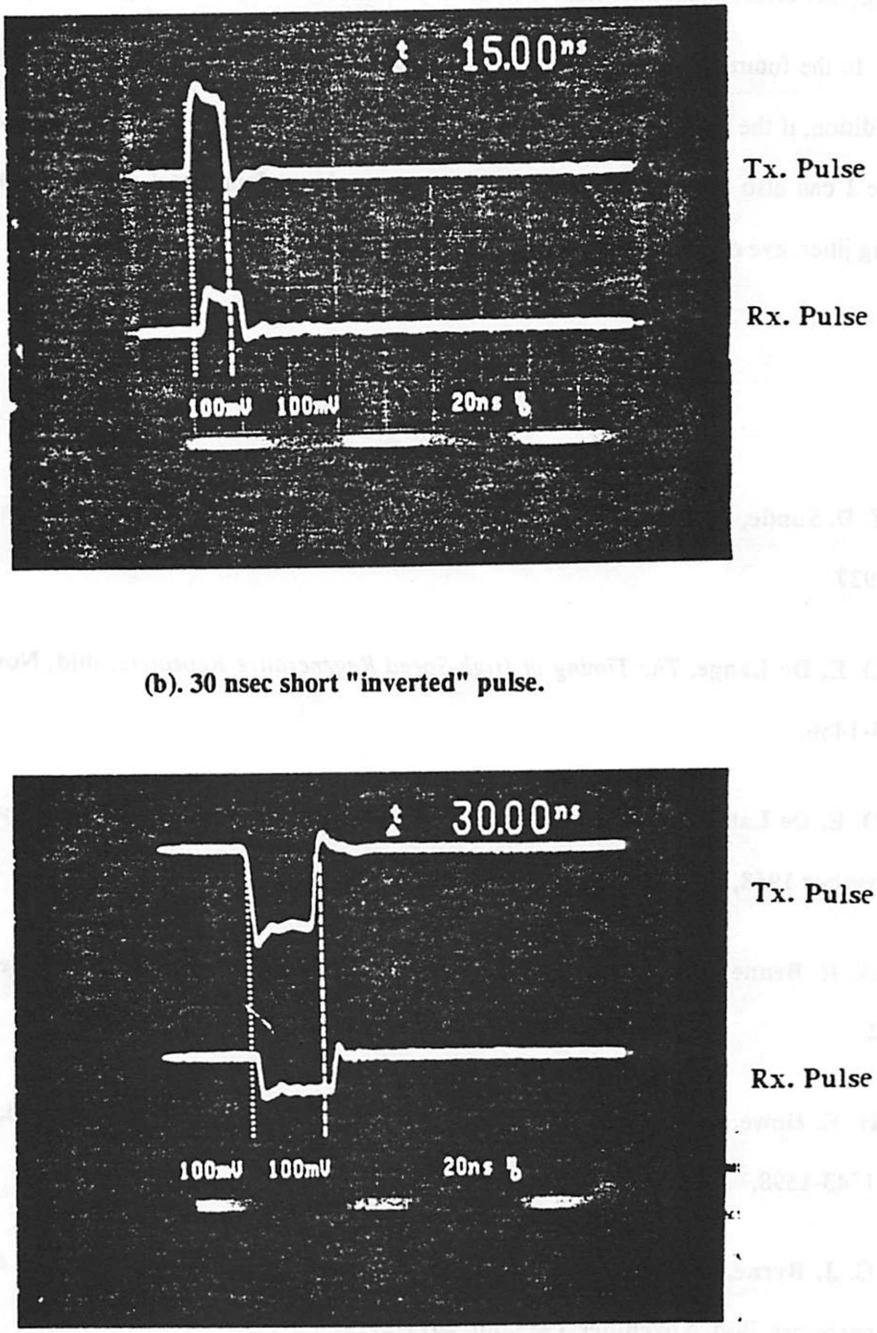


Figure 10. Successful pulse transmission with different duty cycles.



(a). 15 nsec short width pulse.

(b). 30 nsec short "inverted" pulse.

6.5. Future Extension

This chapter provides a preliminary examination of the effect of cascading circuit switches, and a feedback loop testing strategy is also suggested. Basic results for data rates within 50 Mb/s show the timing jitter effect is not important if switches are cascaded without timing recovery.

In the future, if the high speed circuit switch becomes available, this jitter test can be repeated. In addition, if the number of available optical transceivers is large enough, testing using the method in figure 1 can also be constructed. In this case, a pseudo random signal can be sent to the system; the timing jitter, eye diagram, and bit error rate can all be measured.

REFERENCES:

[1] **E. D. Sunde,** ***Self-Timing Regenerative Repeaters,*** **Bell System Technical Journal, July 1957, pp. 891-937.**

[2] **O. E. De Lange,** ***The Timing of High-Speed Regenerative Repeaters,*** **ibid, November 1958, pp. 1455-1486.**

[3] **O. E. De Lange and M. Pustelnyk,** ***Experiments on the Timing of Regenerative Repeaters,*** **ibid, November 1958, pp. 1487-1500.**

[4] **W. R. Bennett,** ***Statistics of Regenerative Digital Transmission,*** **ibid, November 1958, pp. 1501-1542.**

[5] **H. E. Howe,** ***Timing in a Long Chain of Regenerative Binary Repeaters,*** **ibid, November 1958, pp. 1543-1598.**

[6] **C. J. Byrne, B. J. Karafin, and D. B. Robinson, Jr.,** ***Systematic Jitter in a Chain of Digital Regenerators,*** **ibid, November 1963, pp. 2679-2714.**

[7] Janet M. Cooper, *The Effect of Cascading Optical Links Without Timing Recovery for Applications in Local Area Networks*, Master Report, Department of EECS, University of California, Berkeley, 1987.

[8] H. J. Shin and D. A. Hodges, *CMOS Switch for 200 Mb/s Fiber Optic Networks*, to be published.

[9] S. L. Storozum, *Fiber Optic Systems: Practical Design (III)*, Photonics Spectra, pp. 57-63, November 1985.

[10] AT&T, *Assembly Instructions ST^{TM} Series Lightguide Connectors*, Issue 5A, 1985.

CHAPTER 7

Future Work

7.1. Introduction

In previous chapters, both system and implementation approaches to low cost and high performance fiber optic networks have been discussed. To integrate all these approaches, to appreciate all implementation problems, and to justify the proposed high performance and low cost, the realization of prototype networks is very helpful and important.

In this chapter, a very basic TSS network serving 64 Kb/s voice traffic, described in chapter 2, is designed and described. The timing recovery approach mentioned in chapter 5 will also be adopted. Even the network function is very simple, most of the implementation details can be appreciated through the design procedures. This prototype network design presented in this chapter provides a reference for future implementation, and many modifications discussed in chapters 3,4 can be easily extended from this basic design.

Figure 1. A Pure 64 Kb/s Time Slot Switching Prototype Network.

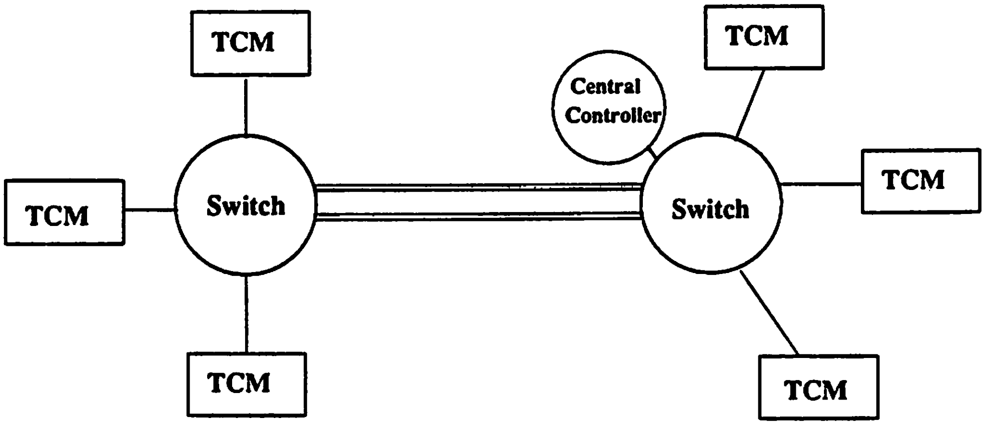
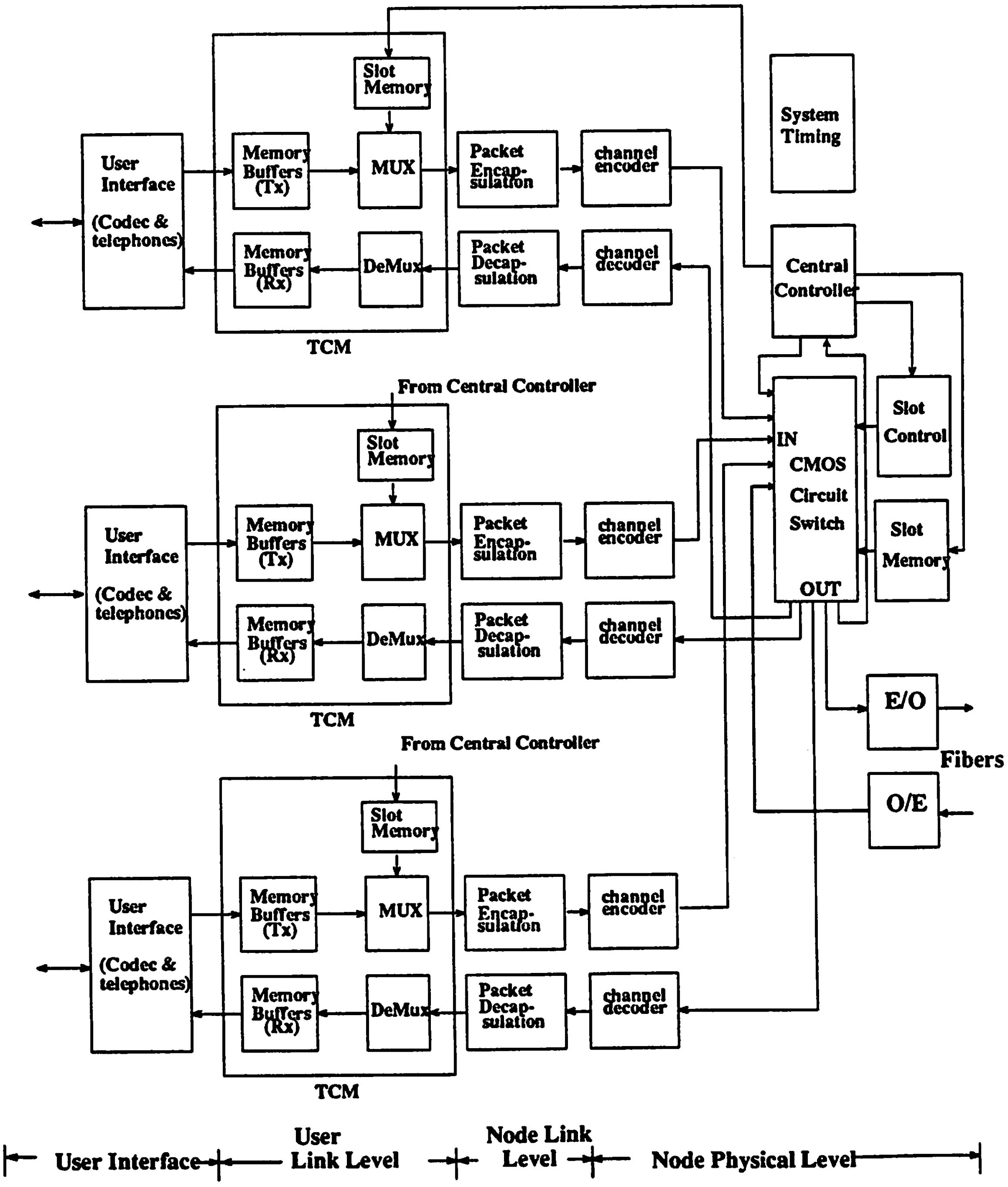


Figure 2. Internal structure of the prototype network in Figure 1.
(a). The right hand side of Figure 1.



7.2. A Prototype TSS Network

Based on the descriptions of TSS in chapter 2, a prototype network topology for pure 64 Kb/s digital voice traffic is shown in figure 1. Two switching nodes are connected by optic fibers, while all telephones are connected to the switches through the TCM (Time Compression Multiplexer). Each TCM occupies one crosspoint pair of the circuit switch, while many telephone users can share one TCM because the fiber optic bandwidth is many times the 64 Kb/s voice rate. The central controller, which provides the frame timing and makes decisions on circuit connections, resides on one of the switching nodes.

Figure 2a shows the internal structure of the right hand side of figure 1: the TCM, the switching node, and the central controller. Figure 2b shows the structure of the left hand side. The reader can compare figure 1c in chapter 2 to see the difference, where the control traffic is merged into the same TSS network as described in section 2.3.3. The internal structures of the TCM, packet en/de-capsulation, channel en/de-coder, user interface, system timing, and central controller in figure 2 will be described below.

7.2.1. The User Interface

The function of the user interface is to serve as a translator between the speech and the digitized bit streams, figure 3. Therefore, it consists primarily of a 64 Kb/s PCM digitization circuit.

64 Kb/s PCM digitization of voice is standard and straightforward: the hybrid for interfacing the telephones, the filters for antialiasing and smoothing, and the PCM codec for the waveform quantization. The digitized bit streams from the PCM codec generally is in serial bits, which is converted to an acceptable parallel format to be stored in the TCM memory buffer.

7.2.2. The TCM

The term, TCM (Time Compression Multiplexer), implies two functions: (1) compressing continuous PCM data into "data packets" and decompressing the data packets into continuous PCM data (see chapter 2), and (2) multiplexing many voice channels into one high speed optic link through the

Figure 2, Continued. (b). The left hand side of Figure 1.

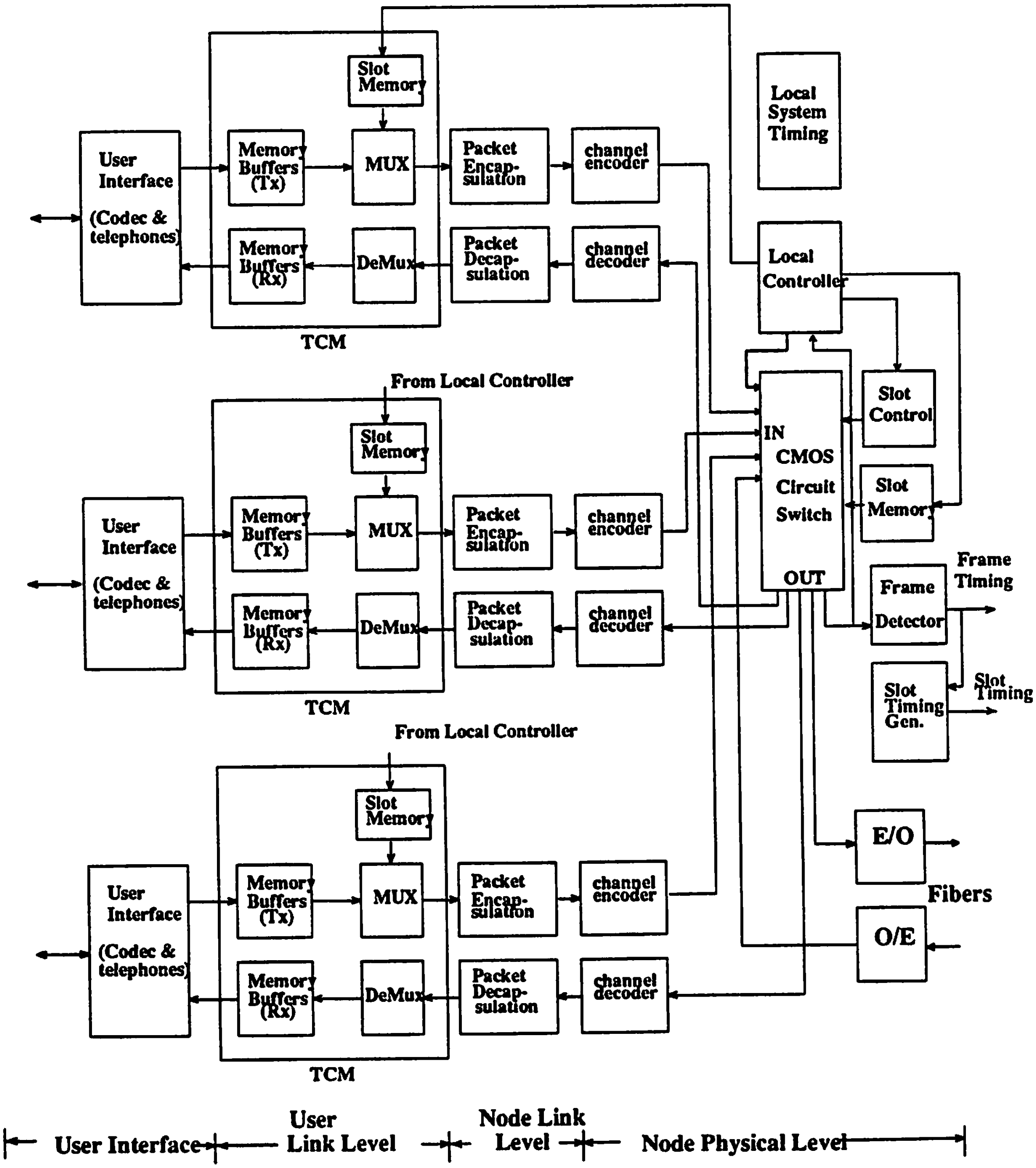
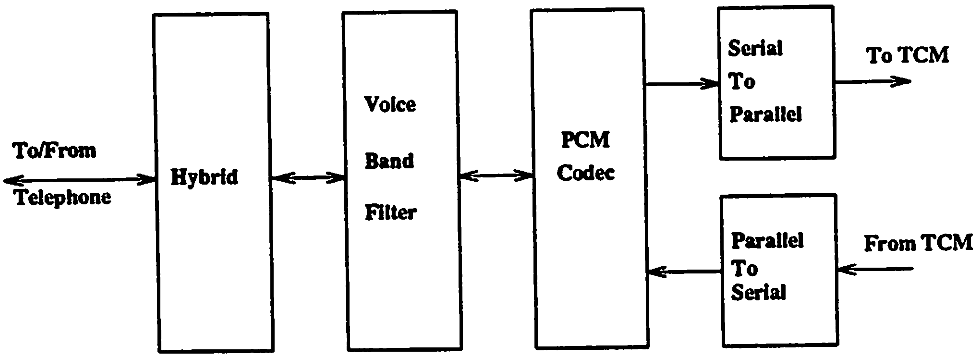
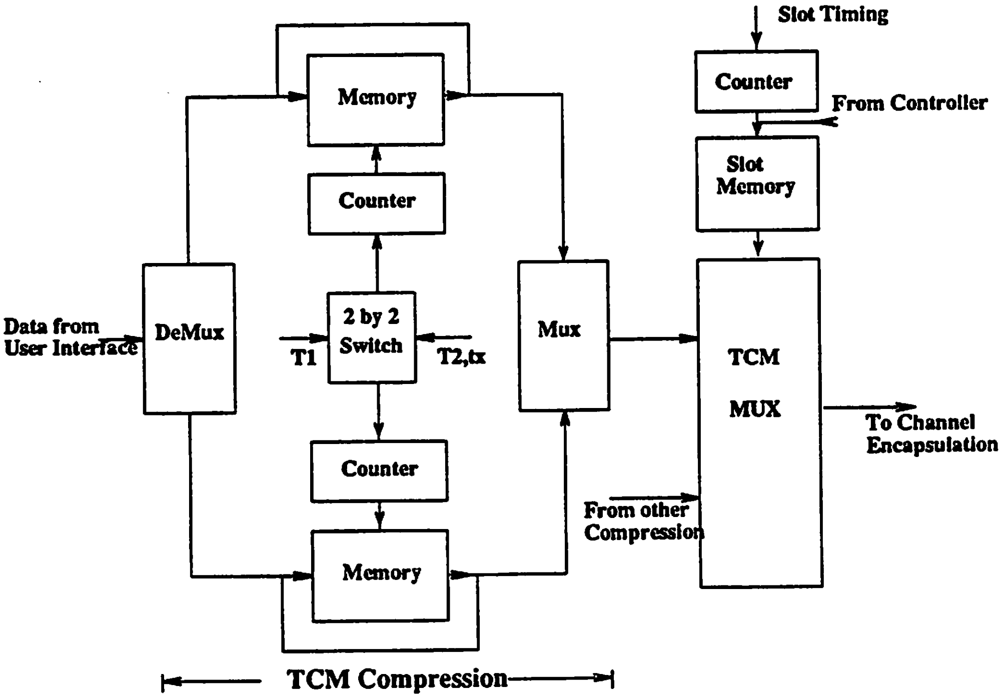


Figure 3. The User Interface.



circuit switch.

Figure 4. The TCM, the transmission part.

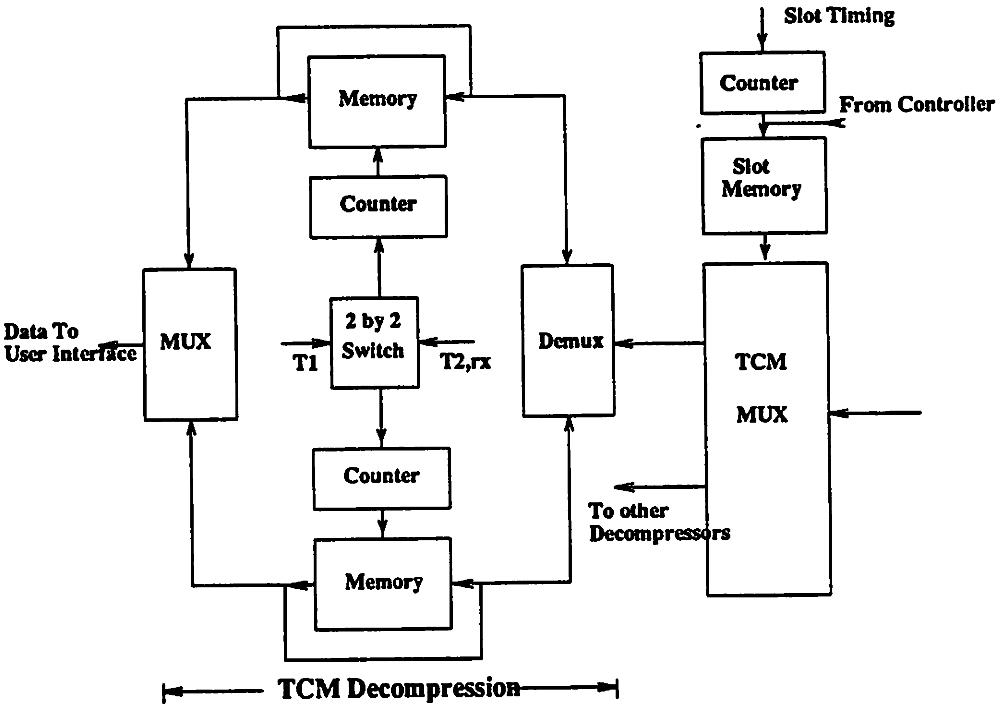


T1: Word Clock from the User Interface

T2,tx: Word Clock from the Channel Encoder

As shown in figure 4, there are two memory banks for the compressor: one for buffering the PCM data, and one for fetching the data at the link speed in certain time slots. The reason for having two memories is that the reading and writing of one memory can not happen at the same time. After each frame, the roles of these two memories will exchange; therefore, additional multiplexing is needed. The counters serve as the addresses for these memories. Since they increment their addresses at different rates (one for buffering data from the user interface, one for fetching data for high speed transmission to the channel), an additional 2 by 2 circuit switch is used to switch these two clocks. The control signal, which controls the switching of the switches, the multiplexers, and the memories, toggles every time frame and is not shown in the figure.

Figure 5. The TCM, the receiving part.



T1: Word Clock from the User Interface

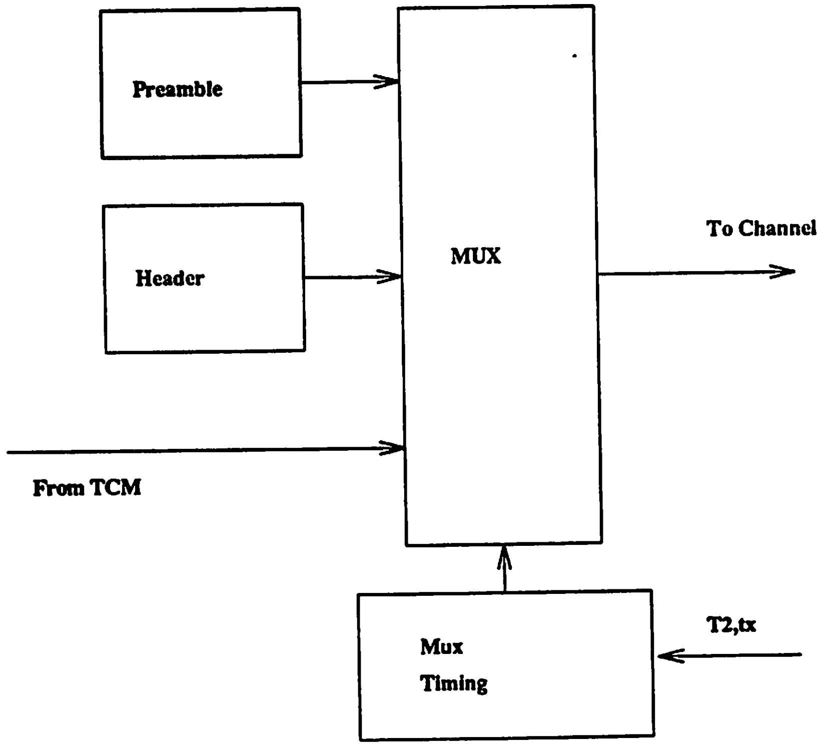
T2,rx: Word Clock from the Channel Encoder

Each voice channel operates at a comparatively lower rate than the high speed link (e.g., 200 Mb/s). To fully utilize the high speed link, many voice channels are multiplexed together. To determine which channel is assigned to each time slot, a slot memory is associated with the multiplexer. The multiplexing information is fed from the central controller. The counter of the slot memory, which increments by 1 after every time slot, serves as the address of the content in slot memory. When new circuit connection information is fed from the central or local controller, the address may also be directly supplied from the controller.

The decompression and the demultiplexer in the TCM, figure 5, do the inverse operation.

7.2.3. The Packet Encapsulation and Decapsulation

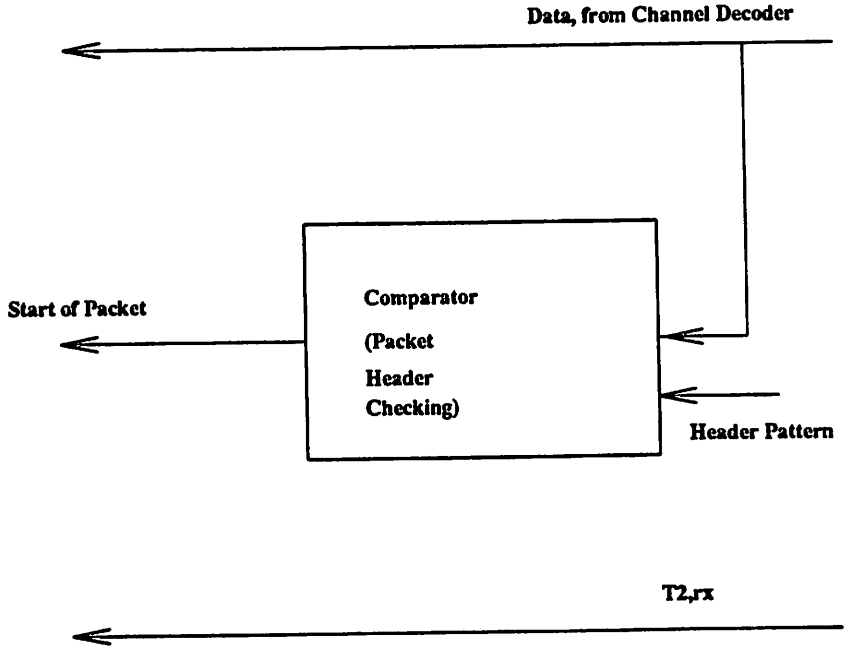
Figure 6. The Packet Encapsulation for Speech.



A data packet itself can not be directly sent to its destination. For the packet switching (e.g. X.21), a preamble sequence, a packet header, the source and destination addresses, acknowledgement bits, the packet length, the sequence check, and other important information for the packet processing are all included with the data in a packet. In our prototype TSS network for voice, due to the nature of circuit switching and speech, and assuming each packet length is fixed, a packet would contain only a slot preamble, a packet header, and the data. The preamble is used for the initial timing recovery, and the packet frame is used to detect the beginning of a packet.

Figure 6 shows the packet encapsulation block, where the preamble, the packet header, and the data are multiplexed to the channel encoder by a proper timing control. Figure 7, the packet decapsulation, does the inverse operation. When a frame header is detected, it recognizes the beginning of the data sequence, and it notifies the TCM to store the incoming bit streams.

Figure 7. The Packet Decapsulation for Speech.



7.2.4. The Channel Encoder and Decoder

The channel encoder transforms packets into a format which will enhance the processing in the receiver. As discussed in chapter 5, the fixed transition code will be used for the purpose of timing recovery in the receiver. The implementation is also discussed in the chapter 5 appendix. After encoding, the data is converted from parallel to serial and sent on the physical channel, figure 8

The channel decoder recovers the bit streams by the timing recovery approach in chapter 5, and at this moment the data is in parallel format A decoder discussed in chapter 5 can be used, figure 9.

7.2.5. The System Timing

In a digital system, correct timing is essential to the proper operation of the whole system. The system timing has two parts: the global timing and local timing. The global timing is the frame and slot timing for every switching node so they can switch on and off synchronously. The central controller has a master clock for frame and slot timings, while other switching nodes regenerate these

Figure 8. The Channel Encoder.

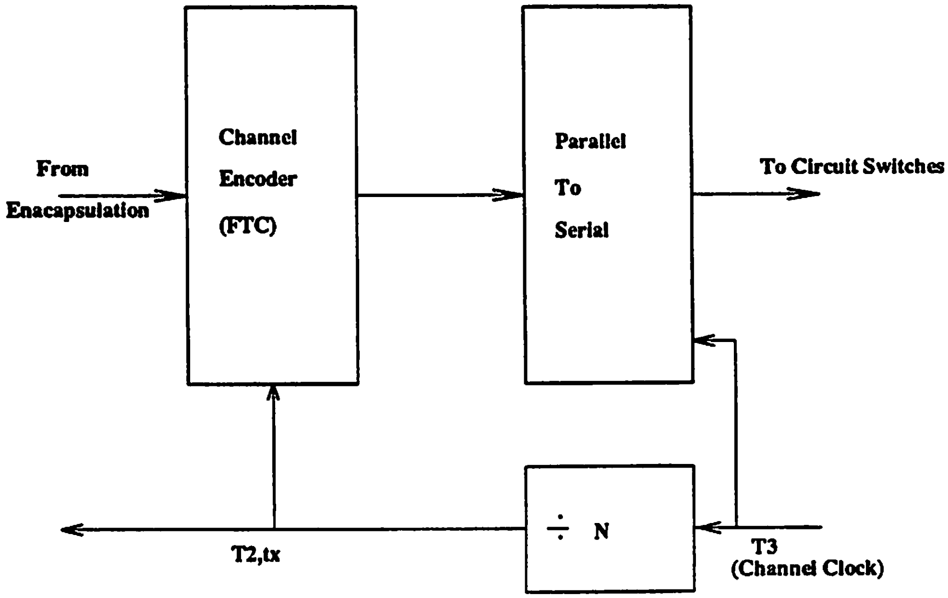
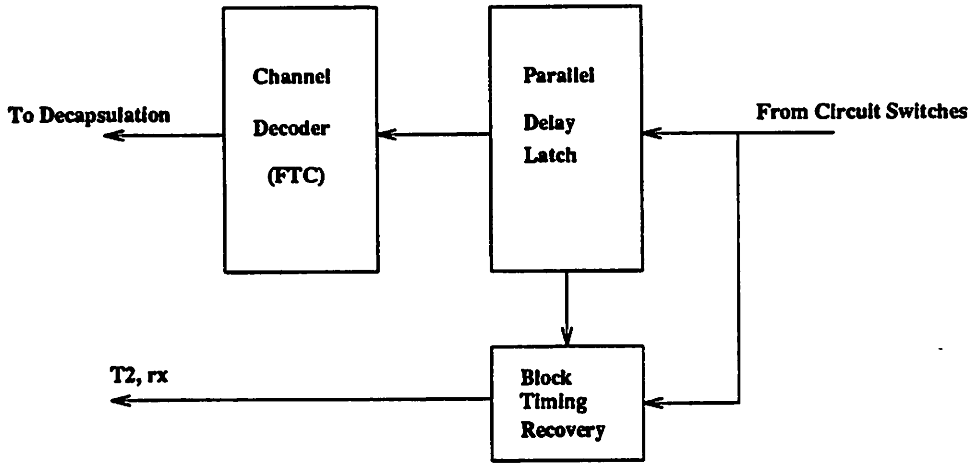


Figure 9. The Channel Decoder.



timings based on the method mentioned in chapter 2. That is, a frame preamble is transmitted by the central controller at the very beginning of each frame. A switching node will recover the frame timing by detecting this preamble and the slot timing by a phase lock loop, figure 2b.

The local timing is used for the internal control of the above mentioned functions. In general, the timing has two separate sets: transmission and receiving. In addition, this timing design is complicated and depends on the physical constraints of the circuit chips used. Consequently, no detailed design is considered in this thesis.

7.2.6. The Central Controller and Control Traffic

The functions of the central controller are to: (1) generate the global system timing, (2) decide the switch and TCM connections, and (3) maintain the network. Figure 10 shows the internal structure of the central controller, where a frame preamble is sent directly to the network at the first slot of every frame to provide the global timing, and the microprocessor determines the traffic switching and monitors the network. The channel codec and the packet capsulation have been described above.

Figure 10. The central controller.

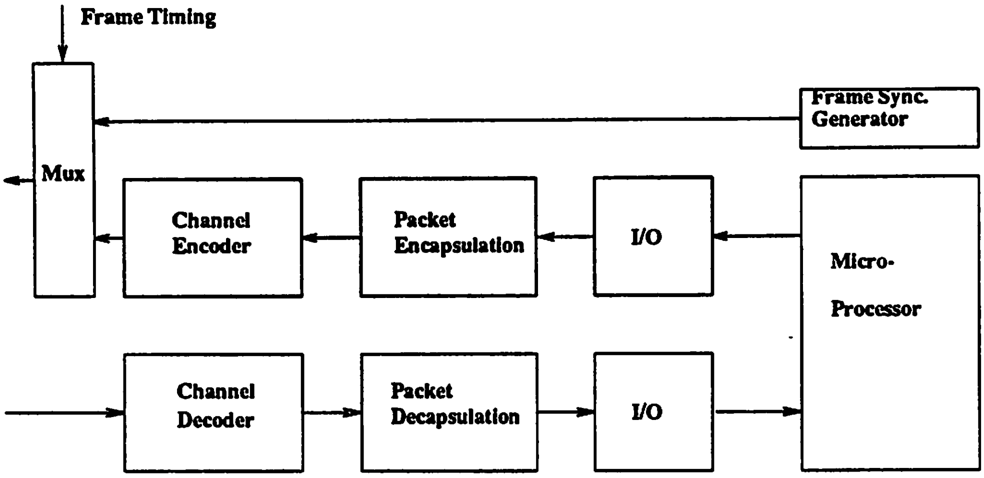
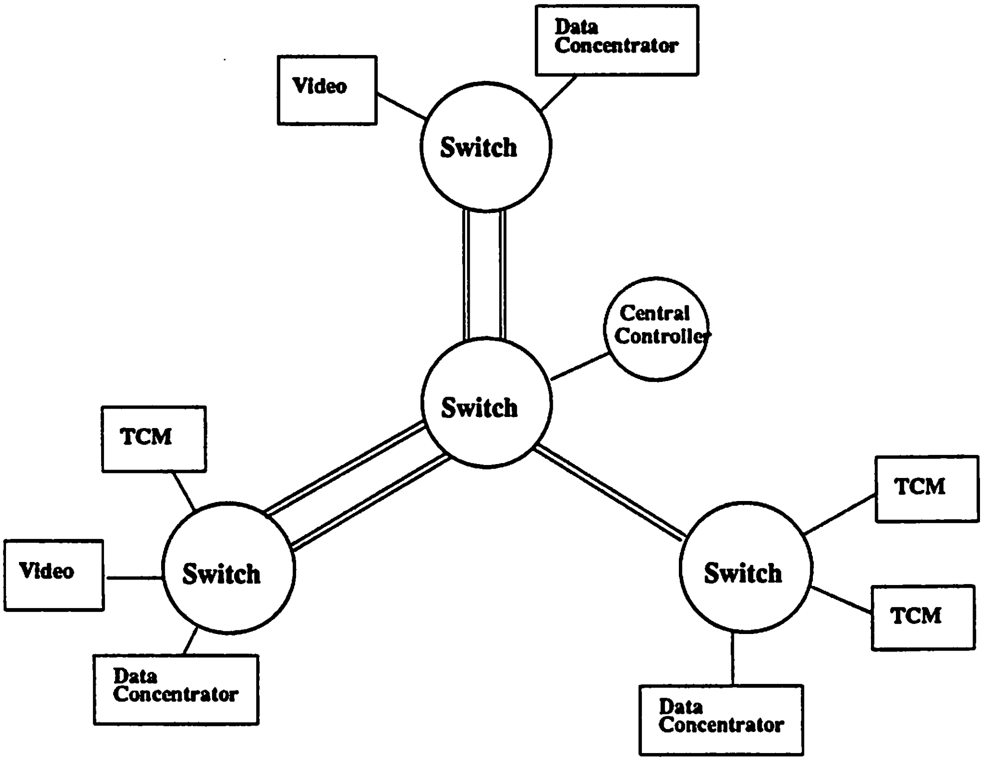


Figure 11. A prototype network for voice, data, and video traffic.



7.3. Modified Prototype Networks

In the last section, a prototype TSS network for pure voice traffic is presented. Networks of more service and larger sizes can easily be extended from this prototype. The principles behind these extensions are based on the discussions in chapters 3 and 4.

7.3.1. A LAN for General Service: Voice, Data, and Video

To provide a general traffic service, we can consider a very simple star network as shown in figure 11. More complicated topologies are not difficult to extend. As noted, there are two links between the switching nodes with video terminals and the central hub. For video service, a dedicated circuit is provided; otherwise, the double links provide higher capacity for data and voice traffic and higher reliability for the system.

Figure 12 shows the internal structure of this simple network, where each block will be the same as mentioned as in the last section except for the user interfaces for the data and video, which depends on the specific terminals used. If CSMA is used together with TSS (see chapter 3), the circuit switches will have two different configurations, as shown in figure 6, chapter 3.

The data packet concentrator merges data packets from different users or directs packets to their final destinations. If this data traffic is served by TSS slot memories and multiplexers will be used, just like the TCM in figures 4,5. If integrated CSMA and TSS is used, the data packet concentrator will merge outgoing packets from data terminals into a queue based on some rules (e.g., FIFO), and direct received packets based on their final destination addresses. In addition, the carrier sense signal (cp. figure 6b, chapter 3) from the switch is checked before the concentrator sends the packets.

7.3.2. A MAN with Skew TSS and Slot-Ring Service

Based on the studies in chapter 4, a metropolitan area network with larger area (e.g. 50 Km) can be implemented using the skew TSS and slot-ring, which are not limited by the larger propagation delay. The prototype network considered will be the same as shown in figure 11, where each optical link is however assumed to have large propagation delays.

Figure 12. An internal structure of the prototype general service network.

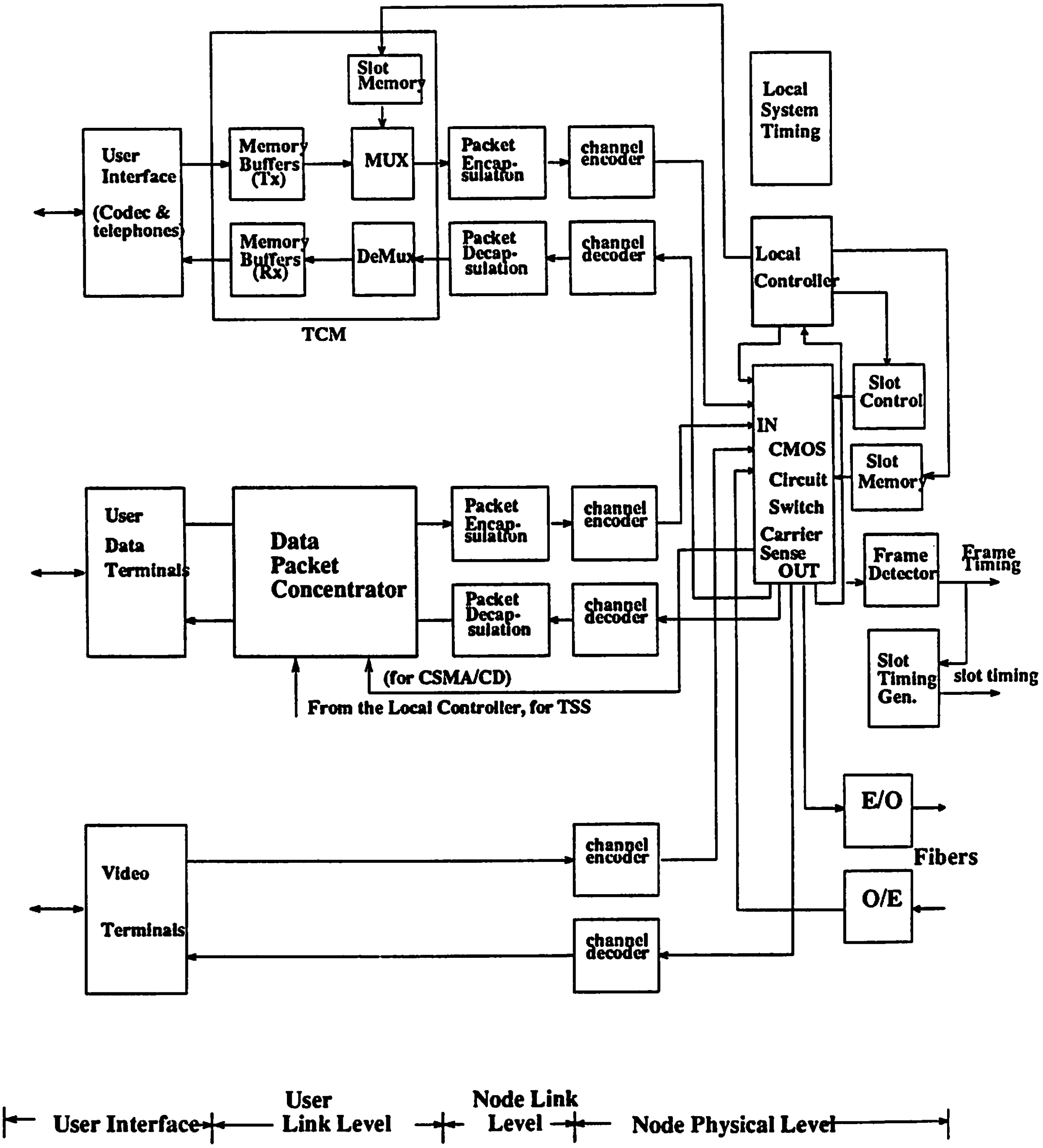
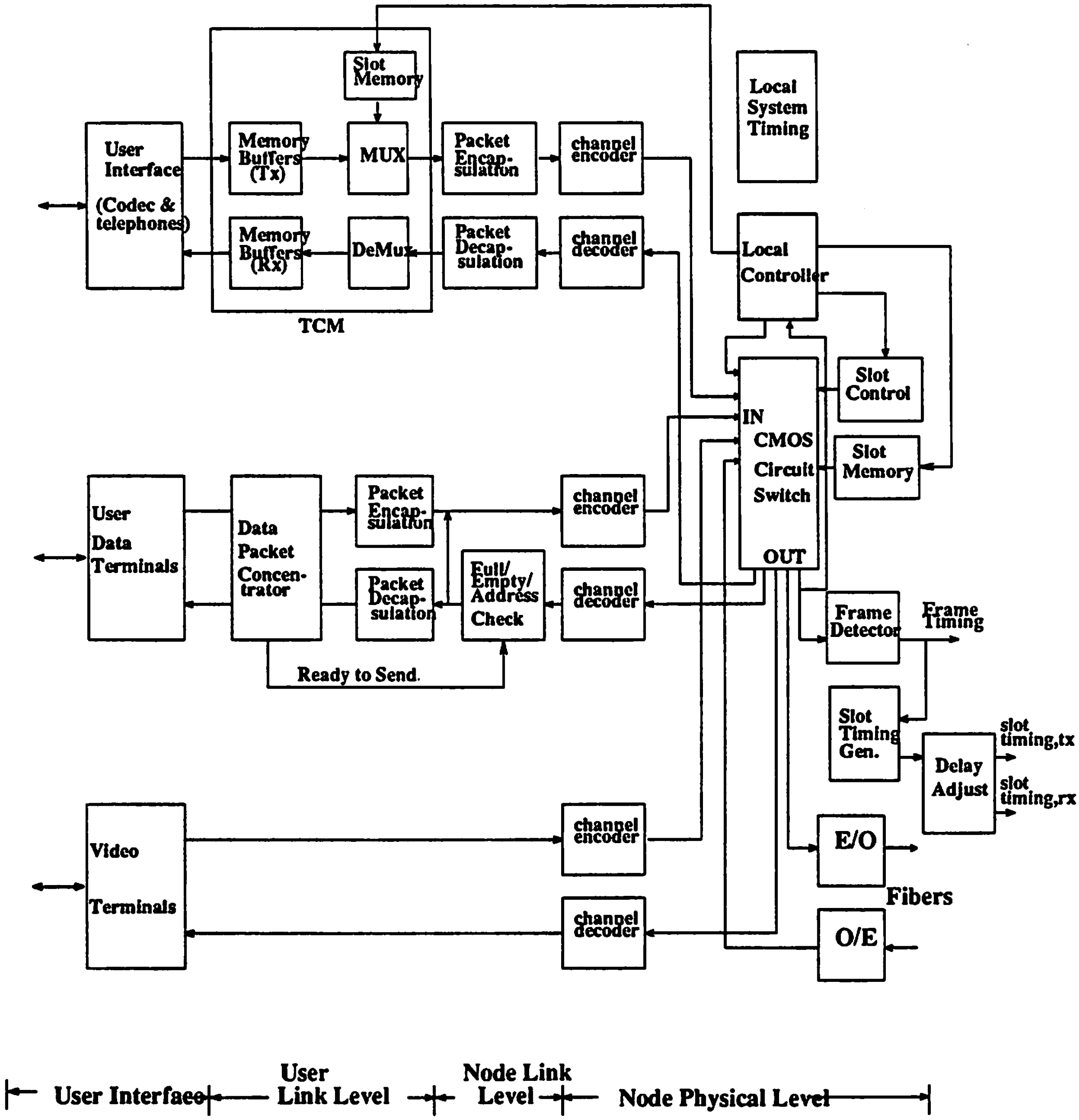


Figure 13. The internal structure of the STSS and slot-ring for MAN.



If the STSS and slot-ring approaches in chapter 4 are used, the internal structure is shown in figure 13. If the central controller is located at the central hub, the frame and slot timing will be adjusted to provide a timing relationship shown in figures 3,4, chapter 4. In this way, a skew time slot schedule can be assigned and the guard time can almost be eliminated.

During the slot-ring operation, a high speed processing circuit is used for empty/full/address checking at the block rate. If a node has packets to send and the received slot is empty, the packet can be inserted and transmitted through the network.

7.4. Conclusion

In this chapter, as the final chapter of this thesis, we explore many network implementations to test and realize the approaches described in previous chapters. The intention of this chapter is to fill an inherent gap between theories and implementations. For the detailed circuit layout and control timing, it would be straightforward from the data sheets and the descriptions in this chapter.