DEPARTMENT OF COMMUNICATIONS

CANADA

MINISTÈRE DES COMMUNICATIONS

VOLUME 1

Introduction To Tîme Shared Systems

- Computer Considerations
- Communications Considerations

by

John deMercado

Terrestrial Planning Branch

Department of Communications

June 1972

DEPARTMENT OF COMMUNICATIONS

CANADA

MINISTÈRE DES COMMUNICATIONS

VOLUME 1

Introduction To Time Shared Systems

- Computer Considerations
- Communications Considerations

by

John deMercado

Terrestrial Planning Branch
Department of Communications

June 1972

Acknowledgements

The purpose of these notes is to promote dialogue within the Terrestrial Planning Branch and serve as a basis for our computer-communication systems implementation program.

The notes are only in first draft form and borrow heavily from the references. They should be read in conjunction with the attached reference papers.

As a revised version is planned the author would appreciate any corrections or omissions in the text that were brought to his attention. He also wishes to thank Messrs. John Harris, S. Mahmoud and Kalman Toth for their valuable contributions.

Miss Gail Widdicombe and Miss Yollande Chartrand typed them in record time from an almost unreadable handwritten manuscript.

Contents

Page

Computer Considerations

Introduction

The ultimate effective use of a computer involves the effective encoding of algorithms in a form so that they can be executed ("efficiently") by computers.

An Algorithm is a list of instructions which in effect specify the sequence of operations (including the operations themselves) which will allow the answer to be computed to any problem in a given class.

A property of algorithms which contributes to their usefulness is that in general the number of operations to be performed in finding the solution to a specific problem is not known a-priori because it depends on the specific problems and is usually only determined during the course of the computations.

The operation of a computer follows this Algorithmic process. Computer operation is governed by the instructions which reside in internal storage and which are interpreted and executed by the circuitry of the arithmetic and control parts of the machine.

These instructions which are usually native to a particular computer are primitive; typically each is composed basically of an operation and one or more operands or modifiers. Instructions which exist in this form in a machine are said to be in machine language.

Thus a computer program or simply a program which is in concept an algorithm and a sequence of machine language instructions can be defined as a meaningful sequence of statements possessing an implicit or explicit order of execution and specifying a computer oriented representation of an algorithmic process.

Statements are strings of symbols (letters, digits and special characters) from a given alphabet. The set of rules that govern how these statements are formed is called the syntax and the operational meaning, the semantics.

LANGUAGES

A *language* is then collectively, the *alphabet*, *syntax* and *semantics*.

A *machine language* has an alphabet of internal machine codes, a primitive syntax of operations, operands and modifiers and semantic rules determined by the circuitry of the machine. Machine languages as a result of many inherent properties are not suitable for direct use by humans in problem solving and *programming languages* and *operating systems* have been developed which significantly contribute to the ease with which humans can program computers. Several programming languages have been developed that have the following advantages over machine languages, namely they

- are more suitable for human use
- are associated in some sense with the problems under consideration
- are designed to facilitate the programming of computers by those wishing to solve problems.

The attractive feature of programming languages is the fact that computer programs written in this form can be translated to machine language by another program running on the same or different machine. Thus the use of programming languages and translation programs allows the same machine to process programs written in many different languages, provided of course that a translator program has been developed for each language.

The "simplest" type of programming language is known as *Assembler language*. This language provides commands or operations that are very similar to the machine language of the computer being programmed. The syntax used with assembler languages provides that each line of coding is composed of two basic fields:

- the statement field
- the identification-sequence field.

Thus a statement in assembler language consists of one to four entries in the statement field, namely left to right: location, operation, operant and comments.

Because there is a close correspondence between assembler and actual machine language, the translation process is not a difficult task.

FORTRAN and COBOL are examples of procedure oriented programming languages, and are more easily used by humans than assembler or machine languages. They are, however, not similar in syntax to that accepted by the circuitry of the computer on which the programs are to be run, they possess sophisticated syntax and semantics and this makes their translation to machine language a very involved and complex process.

Problem oriented languages are languages that are specialized for the description of particular problems. These languages often use whole sentences and words from the vocabulary of the user, and GOGO, and ECAP are examples of these languages.

META PROGRAMS

The concept of a meta program parallels many of those concepts that have been fruitfully employed in other areas in mathematics, e.g. in functional analysis.

Simply speaking a meta program is a computer program which operates on programs. If the output of this operation is a program then the metaprogram is called a translator. Thus it is the translator that performs the map of FORTRAN to machine language.

A translator is called a compiler when it maps programs of one language into programs of another language. An assembler is a special type of compiler that operates on programs whose statements are primitive and independent. An interpreter is again a special type of translator, which in operating on a given program produces another program only as an intermediate step and discards it after it has executed it.

The execution of an interpreter usually proceeds as follows:

- each statement in the given language is first translated to an intermediate language. This statement is then <u>interpreted</u> in this intermediate language and then is executed. Only the results of the executed statement are retained. It follows that statements from the given program are selected for processing by the interpreter in a sequence determined by the execution of preceding statements.

The objective of an assembler metaprogram is to produce machine language programs. Although the majority of compiler metaprograms also produce machine language programs, they are not so restricted by definition. In fact, many compilers do produce assembler language programs. The choice turns out to be one of economics rather than one of basic philosophy.

Most computer applications require a combination of mathematical, file processing, and retrieval techniques, and these capabilities are usually well ingrained in procedure-oriented languages. Metaprograms on the other hand involve character manipulation, table construction and searching, and various types of error analysis. For this purpose, most procedure-oriented languages are either inconvenient or inefficient for use as programming metaprograms, and many metaprograms are coded in assembler language. Several problem-oriented languages, more suitable to the requirements of meta-programming, have been developed and are generally available. For example SNOBOL is a character manipulation language.

<u>EFFECTIVENESS OF COMPUTER INSTALLATIONS</u>

The effectiveness of a computer installation is measured by its ability to satisfy needs for computation. To a large extent, effectiveness is influenced by the speed and configuration of the available computer. To an equally large extent, however, it is affected by the manner in which the hardware system is used and by how the installation meets its demands forservice and for programming. The hardware system, along with the programmed facilities available for using it, is termed the computer system, and affects the work situation in the following ways:

- by its ability to keep all the system's hardware facilities as busy as possible;
- by the extent to which the computer system is available for program development, program debugging, and for priority processing in addition to the normal production workload;
- by the versatility that the system provides for interchanging input/output device types and for accepting varied file organizations;
- by the reliability of the system so that it is available upon demand.

In designing a time shared system three criteria must be satisfied, namely the designer should attempt to

- maximize the use of the system's resources
- reduce the complexity involved in preparing a program for execution on a computer,
- give the user increased control over the way his program is processed by the computer system.

OPERATING SYSTEMS

An operating system discussed briefly below is an integrated set of control programs and processing programs designed to maximize the use of the system's resources and to reduce the complexity involved in preparing a program for execution on a computer, and at the same time the operating system must meet the above criteria.

Operating systems also attempt to maximize the use of the hardware resources while at the same time provide a range of programmer services.

The operating system operates under the direction of user-prepared control cards, which enable it to call any required program or language translator and to pass automatically from job to job with minimum delay and operator intervention. This is accomplished by designing the system so that it can stack jobs for continuous processing, thereby reducing the setup time between jobs. User communication with the computer is through the operating system.

CONTROL & MANAGEMENT

Control programs monitor the operation of the entire system, supervise the execution of the processing programs, control the location and storage of data, and select jobs for continuous processing. The set of functions performed by the control programs are separated into three categories:

- System Management functions
- Data Management functions
- Job Management functions.

System Management handles all actual input-output operations, interrupt conditions and requests for the allocation of system resources. It also controls the execution of the processing programs.

Data Management handles the cataloging of data sets, the allocation of space on external storage devices, the building of program libraries, and the coordination of input-output activity between problem programs and the necessary system management functions.

Job Management reads and interprets user control cards, readies programs for execution, monitors the execution of processing programs, and provides a variety of system services.

Collectively, it is the control programs which tie the rest of the system together and permit the maximum use of a central computer facility capable of servicing many users.

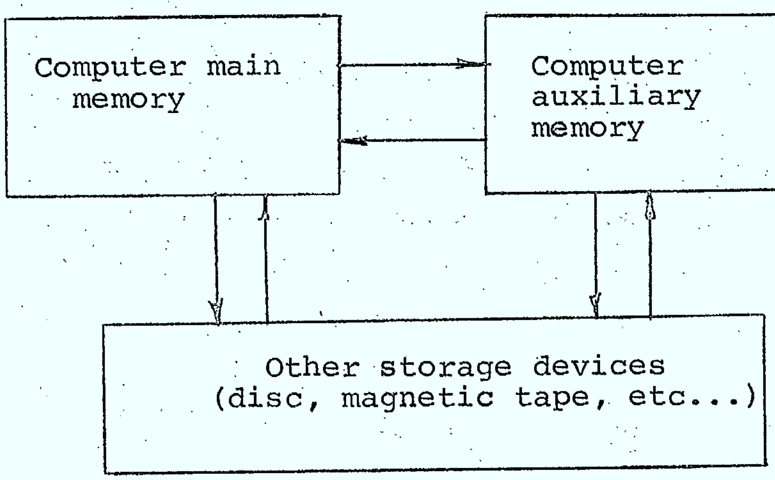
Processing programs consist of language translators, system service programs and user-written problem programs. The programmer uses these programs to specify the work that the computing system is to perform and to aid in program preparation.

System service programs include the utility programs available with any system as well as standard software packages such as the Scientific Subroutine Package for the IBM-360. A comprehensive operating system also supports the generation, storing, and sharing of user written programs.

TIME SHARING

The primary purpose of time-sharing is to provide many users with simultaneous access to a central computer. Although time-sharing systems are usually characterized by remote terminals and a conversational mode of operation, the concept is more general and applied to a variety of operating systems as well. The notion of a time-sharing system is nothing more than a comprehensive operating system with extended features to cover storage allocation, program relocation, program segmentation, job scheduling, and the sharing of system resources. Time-shared operation of a computer system permits the allocation of both space and time on a temporary and dynamically changing basis. Several user programs can reside in computer main memory at one time. while many others reside also in computer auxilliary memory, and temporarily on other auxiliary storage, such as disc or drum (see figure 1). Computer control is turned over to a resident program for a scheduled time interval or until the program reaches

Two memory Level Computer


Computer main memory
Computer auxiliary memory
Other storage devices (disc, magnetic tape, etc...)

- In most computers we only have the computer main memory and other storage devices.
- A growing number of modern computers combine large, relatively slow-access magnetic core memory (auxiliary memory) with much smaller capacity, faster and relatively costlier semiconducts memories (main memory).
- Programs are executed only when their pages reside in the main memory.

Figure 1

a delay point (such as an I/O operation), depending upon the priority structure and control algorithm. At this time CPU (central processing unit) control is turned over to another program. A non-active program may continue to reside in computer storage or may be moved to auxiliary storage to make room for other programs and subsequently be reloaded when its next turn for machine use occurs.

Management of a complex operating environment such as this requires a sophisticated set of control programs of the types covered earlier. Obviously, sophisticated control programs require a sizeable investment for development and use overhead computer time during execution as well. The widespread acceptance of time-shared systems would indicate that ample justification exists. Time sharing exists for many reasons, and these reasons differ from system to system. Basically, these reasons fall into the following categories:

- to maximize the use of the system's facilities;
- to reduce the turnaround time for small jobs;
- to give the remote user the operational advantages of having a machine to himself by using his think, reaction or I/O time to run other programs in the CPU;
- to enable remote users to enter data into and receive data from the system via communications facilities;
- to provide an environment for real time or "demand type" processing and
- to reduce problem set-up and operational times and to minimize the complexities involved with these functions.

CPU ALLOCATION CONCEPTS

The most important resource in a time-sharing is CPU time, without which no job could be serviced. Because of its importance, time-sharing systems tend to be classified on the basis of how this valuable commodity is allocated. Five relevant concepts or conditions for scheduling and CPU allocation are:

- sequential;
- natural wait;
- time slicing;
- priority;
- demand.

These concepts, obviously are not mutually exclusive and are used in various combinations in the different types of time-sharing systems. The techniques are defined in the following paragraphs; these are used extensively by all practioners in the field.

Sequential The system initiates units of work depending upon their time of arrival in the input stream. Once initiated, a unit of work ties up the entire resources of the system (even though it may not use them all) and runs until completed.

Natural Wait A natural wait is a condition which causes a particular unit of work to be unable to use the CPU for a period of time (e.g. an input-output operation). Most units of work have natural waits imbedded in them. When this condition occurs, the unit of work that is next in sequence is given CPU time.

Time Slicing A time-slicing algorithm recognizes that a certain (dynamic) number of programs must be serviced within a reasonable period of time. To this end, the system maintains a cumulative log of its resources (such as time) used by each program. When a threshold level is exceeded, the program, being executed, is forced to stop and wait while other programs have a chance to use the CPU.

Priority As a unit of work enters the system it is assigned a priority. The program with the highest priority is given use of the CPU. If it is unable to use the CPU, CPU control is given to a program with lesser priority. If at any time, the program with highest priority needs the CPU, it is immediately given this service.

Demand A unit of work is initiated immediately upon request.

From these definitions, an obvious conclusion can be drawn. Time slicing is more restrictive than the natural wait concept, thus time slicing could be imbedded in a system which uses a natural wait scheduling algorithm, while the reverse is not generally true.

Comments on MIS Systems Design Philosophy

Two problems face any student of MIS. First, there is no unified body of literature and/or theory on this subject and second, a lot of assumptions have to be made about the decision-making process. The field is now as far as any successful systems are concerned still in the growing stage.

Major successful applications of MIS systems to date have been in banks, some industrial companies (Xerox, IBM) and Government. As total systems, most have failed. Some parts have worked very well (ie. Accounting information for decision-making purposes), but others have not been as useful. One of the most encountered reasons of failure has been that the system has not been used. A multiplicity of reasons can explain this behaviour. It is quite important to note that most failures have not been for technological reasons, but for some mysterious man-machine interactions.

It is quite interesting to note that the only allegedly-successful system has been the one Xerox developed for internal use. People contacted at this company, did not volunteer much information other than to say that the system was built up (ie. starting with one application and building from there).

REASONS FOR FAILURES

There are three kinds of decisions made:

(1) Long-term (policy) decisions.
(2) Project decisions (Short-term, but involving committee work).
(3) Every-day decisions.

Also, it is proven fact that managers have a natural aversion for written material, and try to do most of their work through verbal communication channels (telephone, friendly chats, scheduled and unscheduled meetings, etc.). Also, in most decisions, there is always a certain amount of "hunch" involved.

These facts give us some idea, of some of the ways in which a system might be built. These are

a) If the system is going to help the manager in making every-day decisions, it has to be a fast and accurate system ("on-line").

b) For policy decisions (like budget), an "off-line" system is very suitable.

c) For project decisions, a combination of both.

d) Since the manager deals almost exclusively in a verbal manner, the system should be a tool which is very fast, and swift (for example, to be able to answer a question instantly at the terminal).

e) Management is also a status symbol. Managers hate to interact with machines and to type. They are "above this", the system should then be conversational and minimize man-machine interactions, from the "man" point of view. It should also be easy enough to understand, so that a secretary can interact with it. These have been the major drawbacks of existing systems.

f) The system language should be very close to English and allow errors.

The Department of Communications system has a design philosophy which takes into account the above considerations.

OBSERVATIONS

Perhaps, the best way to develop a MIS system would be to identify what managers need and in which context they make decisions. Then, to think of all those decisions which can be helped by a computer-based MIS. This identification process can be done by questionnaires, interviews and close scrutiny of typical days of a given manager. The system should then be set-up for these applications, but with the added flexibility to build all additional features without major hardware changes to the system. I might add that technically this is the greatest challenge. An easily-readable user manual should be distributed to all potential users of the system, coupled with a basic training course.

A continuous feedback system should be incorporated, so as to ensure that the system is evolving in the same direction as management is. An important fact about managerial activity is that it is always changing to adapt to new challenges. The MIS system should also be evolving in a dynamic fashion.

One way of getting feedback could be through a user committee, which got together periodically to discuss the system. Another mean of feedback could be through a once-a-week report, asking questions about the effectiveness of the system and how users relate to it. These ways of getting feedback could be coupled with a once-a-year study of managerial work and decision-making activities compared to the previous study to see how it has changed.

ANALYSIS OF MIS SYSTERM PERFORMANCE

What means and criteria should be used to measure the efficiency and effectiveness of a proposed system? Unfortunately, an MIS incorporates features which are not easily quantifiable.

Obviously the first criteria would be those developed for traditional systems and which include queuing theory, statistical packages and delay-time analysis. In any good text book, a good description of each of these can be found. (see, for example, (1)).

But other measures of performance must be investigated. We must try to measure the unquantifiable. One of these could be to find out, on a given day, for what purposes the computer was used and with what results.
It would be a matter of each user keeping a log of the times he used the system, whether he was satisfied and what kind of assistance he received.

REFERENCES FOR MIS SYSTEMS

Each article will have a number (1 to 5) to give its relative treatment of MIS.

(1) denotes a closely-connected article or book,

(5) denotes only a very general connection.

BOOKS

The Evaluation of Information Services and Products; by: D.W. King and E.C. Bryant. -1-

Data Bases, Computer and the Social Sciences; by: R.L. Bisco. -5-

The Design of the Management Information System; by: D.G. Matthews. -5-

Management Information Systems: Progress and Perspectives; edited by C.H. Kriebel, R.L. Van Horn, and J.T. Heames. -1-

ARTICLES

A)	- Datamation	
	- The birth of Nasdag - March 1, 1972	-5-
	- On the boardwalk - July 15, 1970	-5-
	- Getting Ready - August 1, 1970	-2-
	- Management Sector - August 15, 1970	-5-
	- MIS÷ Data Bases - Nov. 15, 1970	-1-
	- A Fable of our Times - Dec. 15, 1971	-3-
	- De Ludi Natura Leber Serundus - Dec.1, 1971	
(7)	- Display Systems - Nov. 15, 1971	-2-
	- Naked come the time-sharer - April 1, 1971	-4-
B)	-HBR	
	- MIS is a mirage - HBR - Jan-Feb., 1972	-1-
	- Problems in Planning the Information System - M - April, 1971.	-1-
	-Blueprint for MIS - Nov.-Dec., 1970.	-1-
	- Corporate Models: on-time, real-time systems - July-Aug., 1970	-2-
	- At last, Real Computer Power for Decision-Makers - Sept.-Oct., 1970	-1-
C)	- Canadian Data Systems	
	- Let's put Management in MIS - Oct., 1971	-2-
D)	- Data Systems	
	- Management's meditations - Jan., 1971	-1-

Communications Considerations

Communications Considerations

Introduction

The "digital computer" found its first usages in large institutions, and the suggestion is that it will shortly be a household word. Early computers were cumbersome and required very skilled personnel to get them to do anything right. Today computers are designed so that almost anyone can be readily taught to use them, at least in a limited way. Computers now have massive central and peripheral memories and many people can use them simultaneously (via time sharing). It is impractical to have all users physically go to the computer, it is easier to bring the computer power to the user. This is accomplished via communication channels which are at present predominantly part of the telephone network. Some of the factors involved in doing this will now be discussed.

We will only be concerned with communications for those time-sharing systems that are fast becoming a significant factor in the scheme of modern business. These systems usually utilize telephone company facilities - either ordinary switched-voice telephone channels or some private line service leased from these companies.

ELEMENTARY NOTIONS

Digital communications has unfortunately been saddled with terminology from telegraphy; as a result numerous confusions prevail; for example, data transmission rates are now given in antiquated terms, in bauds, which is an old telegraph term representing a basic rate of transmission in pulses per second. The amount of bits of information that can be transmitted in each baud is measured as the number of bits per baud.

Many data-communications systems, primarily the slower-speed systems, use binary signalling and, in these cases, the baud and the bit rate coincide.

Another common term for describing transmission rates in data communications systems is to refer to them in units of characters per second. One reason for adopting characters per second for specifying data-transmission rate grew out of a practice in some system when all of the bits that comprise a character are transmitted simultaneously. The pitfall here is that systems using the bit transmission methods may use a different number of bits, ranging from 5 to 11 to describe different characters. Thus a comparison of various transmission speeds in characters per second can be misleading.

COMPUTATION OF BAUD RATES

The following two examples illustrate the notion of a "baud".

By definition a "Baud" is the transmission rate of a digital signal and is mathematically equal to:

$$\frac{1}{\text{(the time interval in seconds of the smallest signal element present in a digital signal}}$$

Example 1:

A 33 Type Teletypewriter output is 10 characters/sec. each character has $\geq$ 11 signal elements per character (figure 1)

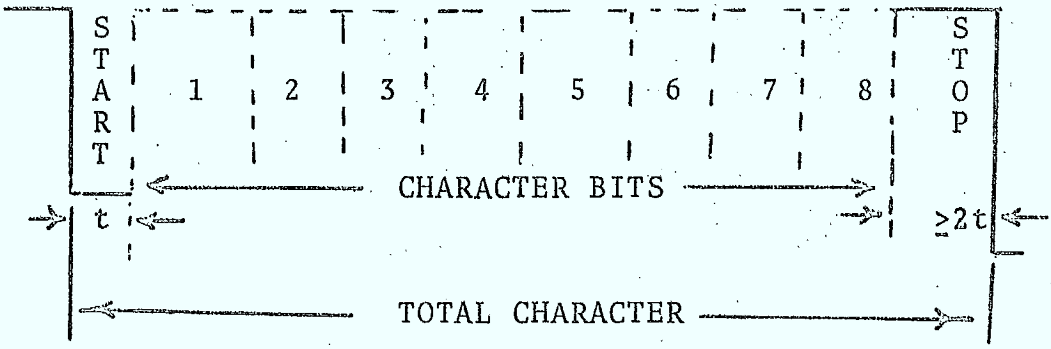


Figure 1

In Figure 1, t is the time interval of the smallest signal element.

$$\therefore\ t = \frac{1}{(\text{Character rate cps}) \times (\#\text{ of signal elements per character})}$$

In this example $t = \dfrac{1}{(10)(11)} = \dfrac{1}{110\text{ sec.}}$

But the Baud rate $= \dfrac{1}{t} =$ 110 elements/sec or 110 Baud.

Example 2:

As a further example, let us compute the baud rate of the multilevel signal shown below in figure 2.

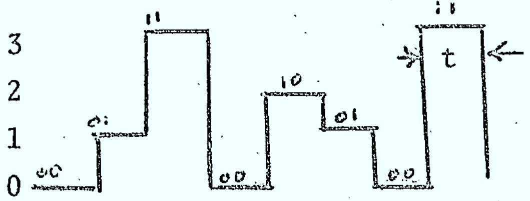


Figure 2 - Multilevel signal

By the definition, the baud rate of the above multilevel signal = $\frac{1}{t}$

therefore: if t = 1 msec. the baud rate would be 1000 baud.

Since the example given is a 4 level signal, there are 2 bits per each signal element, therefore, the information rate is 2000 bits/sec. (assuming all levels have equal probability of occurrance).

The following table shows some baud rates of typical keyboard terminals.

TYPES OF LOW SPEED (_300 Baud) HARD COPY
KEYBOARD TERMINALS AND "BAUD" RATES

Model Number	Manufacturer (or Canadian supplier)	Chars. /sec.	Smallest Signal Element t in sec.	Character in Units of "t"			Baud* Rate $\simeq \frac{1}{t}$
				Start	Data	Stop	
14,15 or 19TTY	Teletype Corp.	6.13	22	1	5	>1.42	45.45
		6.73	20	1	5	>1.42	50
		7.67	17.57	1	5	>1.42	56.88
33 or 35 TTY	Teletype Corp.	6.67	13.64	1	8†	>2.03	73.33
		10	9.09	1	8†	>2	110
IBM 2740 or 1	IBM	14.8	7.5	1	7†	>1	133.3
		15	7.41	1	6	>2	135

* Baud rate ≅ number of signal elements/sec.

Rates on telephone lines are sometimes specified as "slow", "medium" and "fast". There are no universally accepted definitions of these ranges, but typically 300 b/s is slow, 1200 b/s medium and 2400 b/s fast. There is now a growing tendency to shift the slow/fast dividing line upward.

Signals from computers and terminals to each other are generally simple on-off waveforms (called baseband). It is convenient as it eliminates the need for modems if signals are transmitted in the same form as they are generated. This can be done, and is common practice if the receiver and transmitter terminal are linked by short sections of connecting cable. (As in an "in house" system).

The telephone links in Canada have not been designed to carry baseband signals. If data is to be moved from one remote location to another, the baseband signals have to be translated, through modulation, into an acceptable form at the proper frequency for transmission. There are three options as shown in the Figure 3.

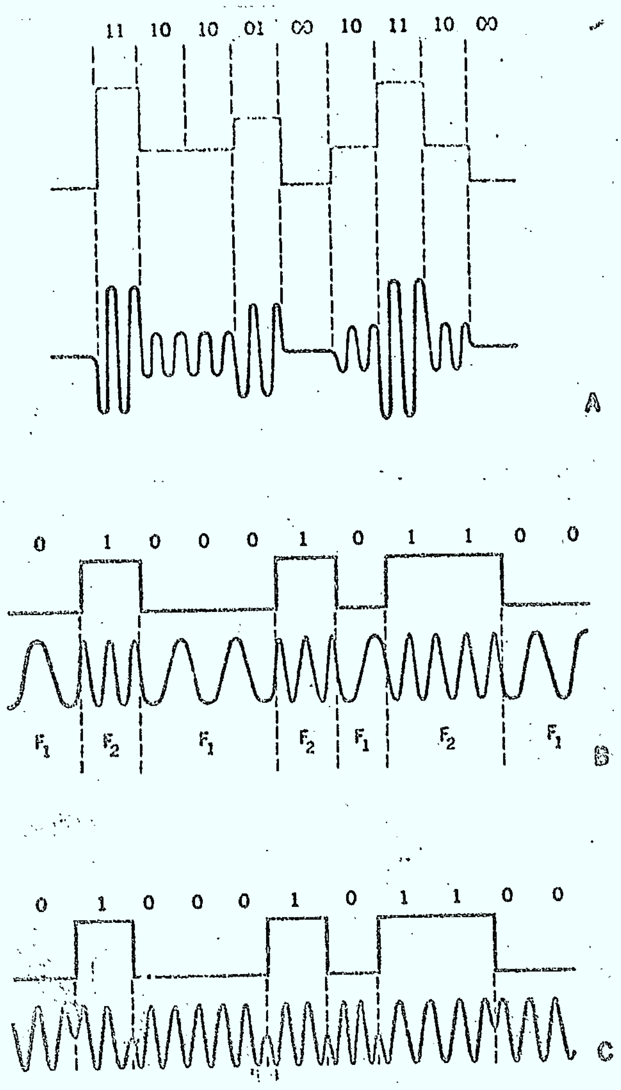


Figure 3

A - when a baseband signal is used to modulate the amplitude of a sine wave carrier this is called in the data field: amplitude shift keying.

B - When the baseband signal is used to control the frequency of a standard signal, this is called in the data field : frequency shift keying. This is the most popular method. Here two frequencies F_1 and F_2 are in effect two tones between 300 and 3000 cps and represent respective 0 and 1.

C - When the baseband is used to control the phase of a generated frequency, this is called phase shift keying.

All of the various methods are used, and amplitude shift keying and phase shift keying have become today the most popular for high speed transmission of data, whereas frequency modulation is the overriding choice for low speed applications.

SYNCHRONOUS AND ASYNCHRONOUS TRANSMISSION

There are two modes in which data is transmitted; namely synchronous and asynchronous. In the synchronous mode a continuous stream of data, which contains information immediately relevant to a message is sent along with some coding as to how the message is to be acquired and reconstructed.

The asynchronous mode is start-stop in nature and either the data stream, the interval between message streams, or both, may occur randomly as shown in figure 4 below.

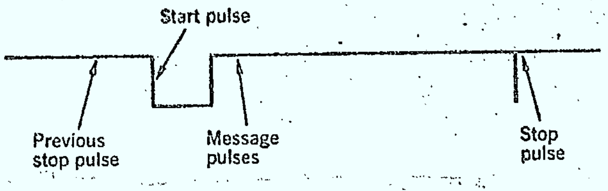


Figure 4: Simple asynchronous receiver keys on start pulse, then samples the remainder of the messages at an internally preset time.

The Asynchronous system contains redundant bits because a start bit and a stop bit are added in each character. Synchronous transmission in which simple framing patterns are applied to long blocks of characters achieved higher message rates. It follows that the synchronous system, lacking this constant reference feature is the choice for highest transmission rates. Apart from the useful fact that asynchronous equipment is usually compatible over a wide range of data rates it is also adaptable to prevailing channel capacities. The telephone switched network, for example, may provide a suitable link for operation at a particular data rate at one time, and at another time the connection may be suitable for only a portion of this rate.

ACHIEVEMENT OF SYNCHRONIZATION IN A TIME SHARED SYSTEM

In a time shared system there obviously must be some way for the communications processor to realize at what instant of time it should extract the message bits. This is achieved by synchronization which has two aspects:

- acquisition; that is, the process of establishing sync, and
- tracking; or the process of correcting for sync drift.

There is no best way to achieve results. The chosen method will depend upon economics, total system characteristics, and technological tradeoff.

In all except the slowest time shared systems, a series of pulses is generated at the terminal to establish sync. This series of pulses continues to be sent until sync is affirmed by say an answer back facility, or sent for some predetermined period of time, if no such answer back facility exists.

There are a number of ways in which synchronization is made. If the system is handling data at precisely defined, regular intervals, then a continuous clock message, transmitted in a portion of the spectrum different from that of the data stream, and filtered out at the terminal, can be transmitted. It is unnecessary for each terminal to contain a clock in such schemes.

Synchronization can be maintained by extracting the necessary clock frequency from the data signal itself. The terminal then keeps its own clock locked to this derived frequency or, in the absence of a transmitted signal, its clock can be idle for a given period of time.

SERIAL & PARALLEL TRANSMISSION

In addition to a choice of synchronization, the data user may choose to transmit either in a serial or parallel mode. Selection usually depends on the original format of the data to be sent, although it also may depend on optimizing channel use.

For parallel transmission, the channel is broken into many subchannels of narrower bandwidth, - that is, frequency-division multiplexed. Typically, each bit of a character is transmitted over a separate narrow band channel. The bit rate of each small channel is reduced by a factor equal to the number of channel segments. In serial transmission, on the other hand, each bit of an individual character is transmitted sequentially over a single channel.

CODES

Data systems transmit binary information and an alphanumeric character set must be generated from arrangements the bits 0 and 1. There are 2^n possible characters that can be formed from a set of n bits. For example, five bits can be used to express 2^5, or 32 characters or numbers.

In general any sequence n of binary digits which is transmitted can be thought of as a code. In general there will be 2^k (where $k < n$) distinct code words that be transmitted using a set of n bits. The remaining $r \equiv n-k$ bits are check bits.

Examples*

The Baudot code is a telegraph code popular in Europe. Each character is represented by five signaling bits. As just explained, five bits, ordinarily, can convey only 32 different characters; by using two of the 32 combinations as a case-shift signal, the Baudot code is expanded to 60 character capability.

The American Standard Code for Information Interchange (ASCII) is rapidly becoming a standard of the data-communications field. This system makes use of seven character identifying bits, which often are supplemented with a parity bit for error checking purposes.

The BCD or binary coded decimal, is a code made up of four bit blocks whose alphanumeric characters can be transmitted in two ways, namely as

- an extended BCD in which two so-called zone bits plus a parity bit are added to the basic four bit block; or
- two blocked together four bit blocks - one for zone and the other for numeric control.

Most magnetic tape units make use of extended BCD computers use a form of blocked together BCD called extended binary-coded decimal interchange (EBCDIC).

* For a further discussion of coding theory and its applications, see my course lectures on Coding Theory, at Carleton University, which are attached as references.

MODEMS, MULTIPLEXORS AND DAA'S - Conditioning the voice network for Data

In order to use a voice network to handle digital signals, additional equipment must be installed. This equipment provides two prime benefits

- technical compatibility between voice and digital modes of operation
- cost reduction by permitting one telephone line to service many actual terminals, thereby reducing monthly communications costs.

The three major pieces of additional equipment required are the modem, the multiplexor and the data access arrangement. The modem, short for modulator/demodulator, is the unit that converts the pulse (digital) signal into an analog signal that can be transmitted over the voice line. At the receiving end another modem converts the analog signal back to pulses (digits) for processing and printout. Modems are now available from telephone companies and from independent manufacturers of data communications equipment.

The multiplexor concentrates the signals from many terminals onto one telephone line for transmission to a distant computer. Another multiplexer at the computer site segments the signals from the transmission line into separate digital sequences, each of which then represents the bits of characters coming from the corresponding remote terminal. The investment in multiplexers is thus reclaimed by not having to pay for separate lines from each terminal to the computer.

The data access arrangement (DAA) interfaces user owned modems to the telephone lines. Its main purposes are to provide specified manual or automatic answer and originate functions and to protect the telephone company equipment and lines from any hazardous situation which might possibly occur through the connection of "foreign" devices. The telephone company leases the DAA to the user. The DAA's used for the Datapoint 2200 CRT terminals in the Department of Communications rent for about $15.00 per month.

The key features of a modem are whether it is high speed or low speed, whether it operates synchronously or asynchronously, and whether it is frequency, phase or amplitude modulated.

Low speed modems, by definition, operate in ranges up to 1,800 bits per second. High speed modems are available in several speeds, the more common ones for industrial and commercial applications, operate at 2,400; 4,800; 7,200 and 9,600 bits persecond respectively. As a rule of thumb, modems cost about $1 per baud.

The real saving in using a high speed modem, provided there is enough data traffic to warrant its installation, is in being able to increase the data throughput without raising line cost. However, everything else being equal, the higher the speed of the modem, the more susceptible it may be to error. And error will incur a cost either in retransmission of the message or in equipment that performs error correction on the received message.

Asynchronous modems do not require or provide timing pulses. A terminal, such as a Teletype unit, depends on the START and STOP pulses to denote the beginning and end of a character.

In their receiving mode, modems must discriminate between the electrical signals that represent 1's and those that represent 0's, and how well they do so is a significant operational consideration. Telephone lines introduce noise and delay distortion, each of which degrades the signal. The error performance of a modem is a rating of its ability to accurately detect signals that are noisy and distorted.

A figure of merit of a synchronous modem is its error rate performance as a function of the signal-to-noise ratio of the received transmission. Rated in this manner, a synchronous modem can be judged on its own performance and not on that of a particular line condition.

For asynchronous modems the performance criterion is called telegraph distortion. Distortion is the time displacement of a 0-to-1 or 1-to-0 transition with respect to a nominal time. Two main sources are amount of misalignment or calibration error in devices in the system, and jitter due to random variations on the line and in the equipment. Generally, asynchronous systems can operate with up to 20% total distortion.

The significance of distortion becomes apparent when full-duplex communication occurs on one pair of lines, which is a most common case. This means that transmitted and received signals are carried on the same line and can interfere with each other. When these signals cannot be separated, distortion becomes large and errors occur. One common way of rating an asynchronous modem in duplex application is by its distortion under conditions of minimum receive level (-49 dBm) and maximum transmit level (0 dBm). A quality modem would have about 7% distortion under this condition, leaving sufficient margin for distortion in the line. The interpretation of this rating is that such a modem can discriminate between a received pulse and a transmitted pulse even though the received pulse is more than 250 times weaker than the transmitted pulse.

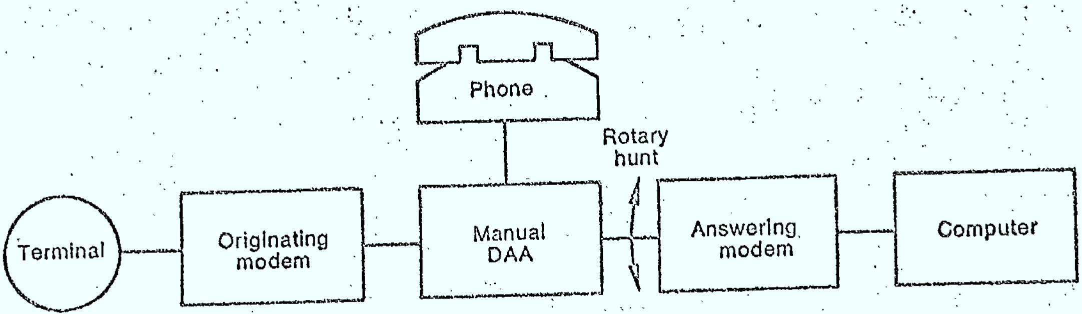
HANDSHAKING

Before a modem pair can perform the main duties of modulation and demodulation, a telephone connection must be established. Control signals from and to modems at each end of the line carry out this modem handshaking operation as will now be described.

Consider a simple installation in which the transmitting terminal and its originating modem are connected to the line through a manual data access arrangement. Through a rotary hunt connection at the telephone company exchange the line goes to an automatic data access arrangement and an answering modem to the computer.

At the terminal, the operator picks up the telephone handset and dials the computer's number. A ringing signal goes through the answering modem to the computer. Assuming the computer is available for business, it sends back a DATA TERMINAL READY signal to the answering modem, which in turn sends a tone signal to the originating modem, an action that turns off the line's echo suppressors that are needed for voice communication but impair digital transmission. The operator hears this tone, and puts the modem on line by lifting the telephone's exclusion key.

The originating modem then sends a tone to the answering modem and a short time later returns a CLEAR TO SEND signal to the terminal. Then the answering modem sends a CLEAR TO SEND signal to the computer. The data link becomes operational, and the terminal can start sending.



Multiplexers perform multiplexing to save telephone line charges, operates in either of two ways: time division multiplexing or frequency division multiplexing. In FDM the available bandwidth of the telephone line, usually from 300 to 3,000 cycles per second is divided into individual frequency slots. Each channel of data is assigned a frequency slot.

Of particular importance in multiplexer selection is the recognition that a communications network may use more than one type of terminal. This means that the multiplexer must be able to handle inputs from terminals with different rates and codes. Ideally, a multiplexer should be able to accommodate Baudot, IBM and ASCII codes and the common terminal speeds of 75, 110, 134.5 and 150 bps. Furthermore, the multiplexer should be transparent to the data; that is, any code and any character set within the code should pass through the multiplexer without modification.

Normally, the total number of bits a multiplexer can process each second can be equal to or less than the rated transmission speed of the modem. Thus many of the same kind of terminals can be multiplexed, or several different types of terminals can be intermixed in the multiplexer. For example, in General DataComm's TDM-1201 multiplexer, eight 300 bps, 10-unit ASCII code terminals can be multiplexed onto a 2,400 bps telephone line. Or, by using special techniques, eleven 110-bps 11 unit ASCII terminals and eleven 134.5 bps 9 unit ASCII terminals can be multiplexed onto the same line.

An interesting extension of the multiplexing function is accomplished by a synchronous combiner. Here, for example, each of two 2,400 bps multiplexers scans its own channels, groups the data streams, and feeds the two resulting streams into the combiner which then outputs 4,800 bps information to a highspeed modem for transmission. However, during night hours one of the groups of channels might not be in operation, so one half of the combiner can then give full service, for example, to a batch terminal operating continuously at 2,400 bps.

Data access arrangements provide a way of connecting user equipment to the telephone line. One type is the manual originate/answer and the more recent type is the automatic originate/answer. Manual DAA's require operator intervention in transferring the line from conventional telephone operation to the modem for data transmission. It contains the necessary electrical controls and safeguards to perform its operations and to protect other equipment. Manual DAA's can originate calls and can answer a call from a remote transmitter. Such units are hardwired to the installation.

Acoustic couplers are also a form of manual data access equipment. An advantage of acoustic couplers is that they offer a portable method of access. The terminal is hardware connected to a modem in the acoustic coupler. A telephone handset is used to dial up the remote terminal or computer. Once the computer is on line, the handset is nested into a cradle on the coupler. The modem in the coupler then converts the digital signals to tones which are picked up by the handset's transmitter and sent over the line.

A more versatile kind of access if the automatic originate/answer data access arrangement. It contains appropriate electrical equipment to automatically make or respond to a call and to convert from voice mode to digital mode, and then revert the line to voice mode when the call is completed. Further, automatic DAA's are equipped to carry out terminal polling with signals originating at the computer. Some of the Data sets available from Bell will now be discussed.

Bell System data sets are classified in five major categories or series. A series is determined by kind of language or transmission mode (analog or digital, either serial or parallel) bandwidth required (narrow N, voice grade V, or wide W) and by its speed.

Within each series, data sets are further identified by the communications facilities required (dial network DDD, or Private Line service); type of set (transmetter T, receiver R, or combined transmitter-receiver T-R); and speed variations expressed in bits per second. The following table summarizes the sets now available.

DATA SETS

Series	Trans-mission mode	Band width	Speed	Type sets	Maximum speed, bps	Facility	Commonly used terminals
100	Serial	N	Low	T-R	150	DDD (TWX) PL	Teletypewriter-like devices
100	Serial	V	Low	T-R	300	DDD PL	Teletypewriter-like devices; low speed CRT
200	Serial	V	Medium	T, R T-R	4,800 10,800	DDD PL	Medium-speed binary quipment; Dataspeed 3, 4; Dataspeed magnetic tape; CRT. Datapoint 2200 - CRT.
300	Serial	W	High	T-R	230,400	PL DDD— (Data-Phone 50 service)	High-speed binary equipment; high-speed facsimile; computer-to-computer; high-speed magnetic tape;
400	Parallel	V	Low	T, R T-R	600 75 cps	DDD PL	Medium-speed paper tape systems; card readers; remote telemetry devices; alarm reporting; audio response systems
600	Analog	V	——	T, R T-R	——	DDD PL	Medical devices; handwriting devices and similar facsimile; telemetry

The following figure 5 illustrates some typical configurations of modems, multiplexers and DAA's. In its simple type of data connection, a terminal at the top left (a) is connected through a private line to the data processing computer at the right. A modem is located at each end of the line. When many terminals located near each other (b) are to communicate with the central computer, their outputs are fed to a multiplexer (mux) which concentrates all the data and sends it out through one modem. Note the saving in modems and lines in this arrangement as compared with the number that would have to be used if separate lines were established for each terminal. However, when the terminals are not close to the multiplexer (c), then additional modems must be used to insure the fidelity of the data signals. In (d), connections are made through a telephone company central office. If the modems are from an independent manufacturer, the installation needs data access arrangements as shown. Experience indicates that when terminals are switched through an exchange, then one output line to the computer can handle three terminals at the input, thus allowing for random online time of each terminal without incurring excessive delays from busy signals.

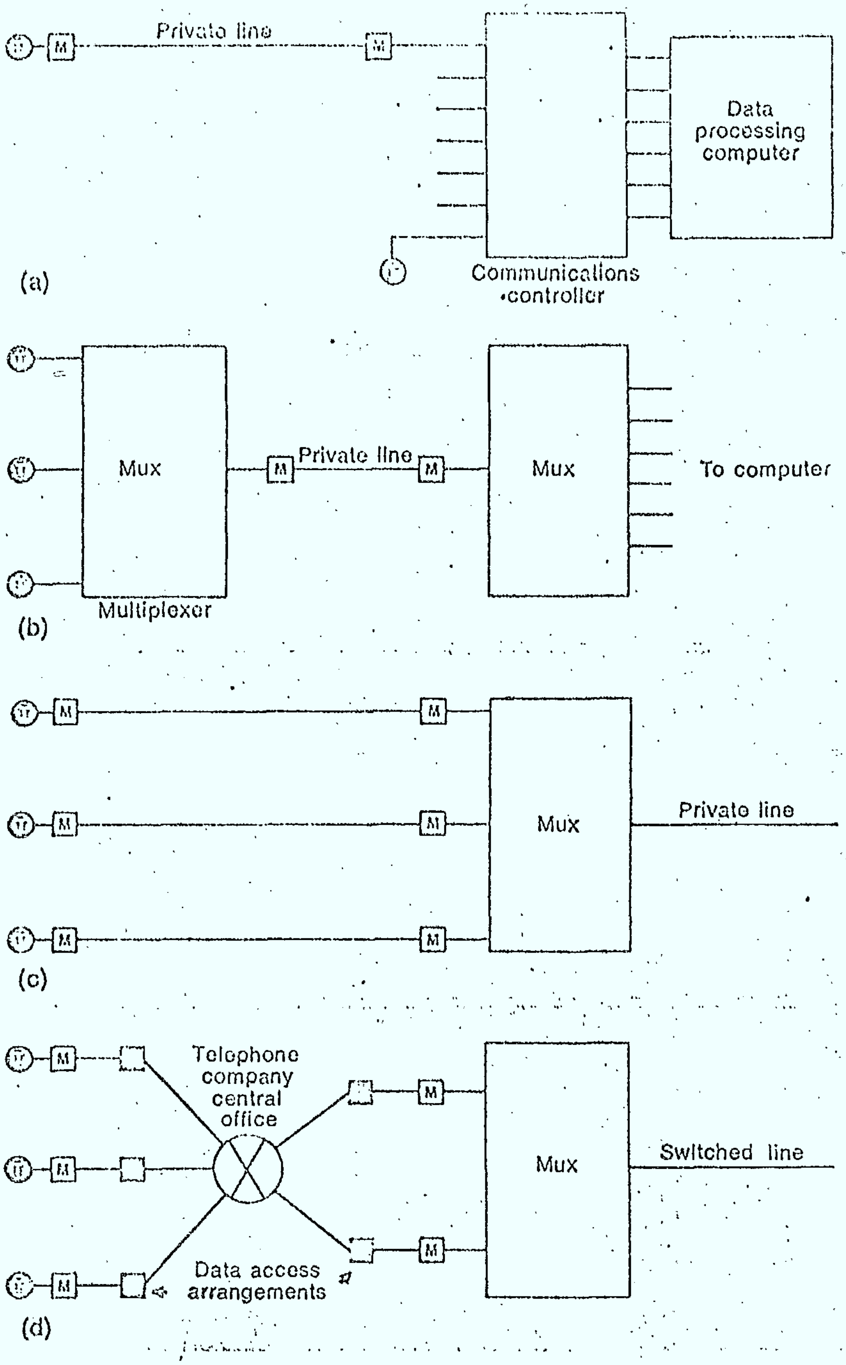
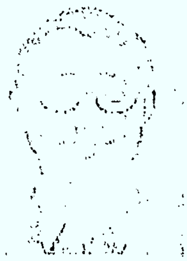

Private line
M
M
T
Communications
controller
Data
processing
computer
(a)
T
Mux
Multiplexer
M
Private line
M
Mux
To computer
(b)
M
Mux
Private line
(c)
M
Telephone
company
central
office
Data access
arrangements
Mux
Switched line
(d)

Figure 5

A TUTORIAL



Sylvain Louchez, B.Eng. (Paris), came to Canada in 1965 for M.Eng. studies at McMaster University. He is presently (since 1967) a member of the research staff of ESE Ltd., Rexdale, Ont. ESE designs and manufactures communications equipment.

Basic communications to help understand data transmission

By Sylvain Louchez

THE BASIC PROBLEM of transmitting intelligence and successfully recovering it from an environment disturbed by noise has given birth to a fascinating and relatively new field — data communications.

A general communications system is shown schematically in Figure 1. It consists of five essential elements, which are briefly described below:

The Information Source can be the human voice, or digital words stored in a computer's memory, a photograph or a television picture, in fact anything that needs to be transmitted.

The Source Transducer is a device that will interpret the intelligence and supply waves or energy related to the input; for instance a telephone microphone, a television camera, a magnetic head in a tape recorder, and a photo electric cell, are all source transducers.

The Channel Encoder can accept the electrical signals from the transducer and transform or condition them to a form suitable for transmission through the selected channel. Any interface or data-set, from the simplest to the most sophisticated code generator, is an "encoder".

The Channel is the transmission medium; it is any type of telephone line or cable or the atmosphere, in the case of radio relay. Most often the channel carries several messages simultaneously, each generated from separate sources.

The Decoder recovers the signals which may have been corrupted by noise in the channel. It can be a simple repeater, or a voice channel separation network, or a sophisticated receiving multiplex terminal, with built-in error detection and correction logic.

The User Transducer receives the proper signal from the decoder; it may be a loudspeaker, a teletype machine or a television set.

Multiplexing consists of combining several different signals, sending them through the same channel, and recovering them at the receiving end.

Speed versus cost

From this point on we shall consider mainly our telephone system as the transmission medium. It is accessible to all, and to date is the most economical mode of transmitting a considerable amount of information. There is, in fact, every indication that more and more busi-

nesses w
data com
responder
moment,
TWX, c
against si
mail. Ho
concern a
by Telex
creasing.
cost of te
If true, w
graph ser

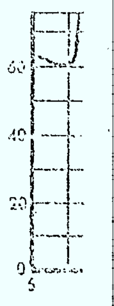
The m
available
data at m
usual to t
telephone
3600, or
per secon
phisticatio
form for
As a resu
tems (ex
speed co
are desig
telephone

The tel
to transm
distortion
gibility, a
to permit
the speak
means th
Hz (cycle
ted; but p
cal. In tra
tortion w
band is m

The comm

Amplit
tics of a
mission c
2. The P
from 200
transmissi
nothing is
4KHz (4 K

Twelve
channels



nesses will be relying on discrete data communications for their correspondence in the future. At the moment, a message sent by Telex or TWX, costs about one dollar against six cents for a letter sent by mail. However, speed is of primary concern and the cost per message by Telex or TWX is constantly decreasing. Some estimates project a cost of ten cents per letter by 1980. If true, we shall all be using a telegraph service within the decade.

The multiplicity of bandwidths available permit the transmission of data at many different speeds. It is usual to transmit, over an ordinary telephone line, 600, 1200, 2400, 3600, or 4800 bits of information per second, depending on the sophistication of the modem (short form for modulator-demodulator). As a result, data transmission systems (excluding of course, high speed computer-to-computer lines) are designed as a function of this telephone channel.

The telephone system is designed to transmit speech without undue distortion. The user expects intelligibility, and sufficient tonal quality to permit recognition of the voice of the speaker. Intelligibility generally means that frequencies up to 3000 Hz (cycles/sec) must be transmitted; but phase distortion is not critical. In transmitting data, phase distortion within a given frequency band is much more important.

The common carrier structure

Amplitude and delay characteristics of a typical voice-grade transmission circuit are shown in Figure 2. The Pass-Band is approximately from 200 to 3200 Hz, there is no transmission at d.c. and virtually nothing is transmitted at and above 4KHz (4 kilocycles/sec).

Twelve such voice frequency channels are frequency multiplexed to give a basic group channel 48 KHz wide by juxtaposition of twelve 4 KHz bands.

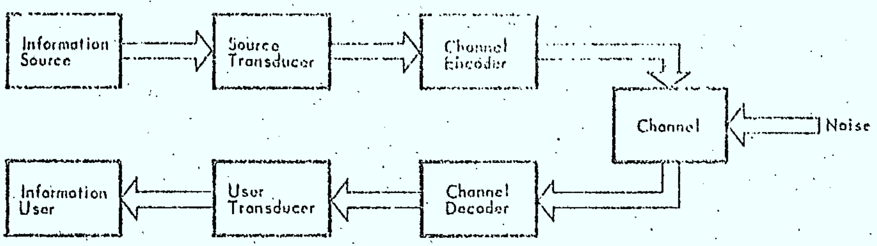


Fig. 1: A basic communication system

Amplitude Modulation translates the frequency of a signal by modulating (mixing) a high frequency carrier with the voice signal. This is shown on Figure 3 and the corresponding frequency spectra in Figure 4. The information contained in either of the side-bands is identical to the original. Only one of the side-bands needs be transmitted: hence the 4KHz voice signal can be sent as a 4KHz bandwidth high frequency signal and be recovered after demodulation.

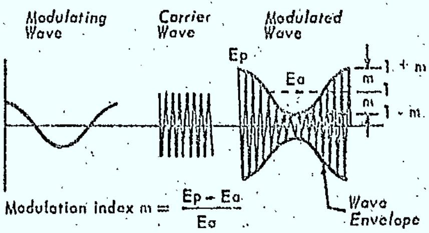


Fig. 3: Amplitude Modulation: By mixing the modulating wave (desired signal) with a carrier wave, the carrier assumes an envelope that faithfully reproduces the modulating wave. This is an additive and subtractive process. For example, if the carrier frequency is 100,000 cps (100 KHz) and the modulating wave band ranges from 200 to 4000 cycles/sec, then two sidebands are set up — one from 100,200 to 104,000 KHz, the other from 96,000 to 99,800 KHz. Since each band carries the same information, one sideband can be eliminated along with the carrier frequency to reduce the bandwidth by half. By re-inserting the carrier frequency at the receiving end and "beating" it with the Single Sideband signal, the original information extracted.

If a large number of voice channels use modulating carriers of different frequencies the resulting side bands will use different frequency bands. They can all be sent together and be isolated again from each other by the use of filters. This concept is illustrated in Figure 5.

The amplitude modulation scheme in this case is the SSBSC: (Single Side Band Suppressed Carrier). Only one side band of the modulated carrier is transmitted over the channel. The carrier is also suppressed, reducing to a minimum the energy carried in the channel (see Figure 6). At demodulation time, the carrier is re-inserted and the original information is recovered.

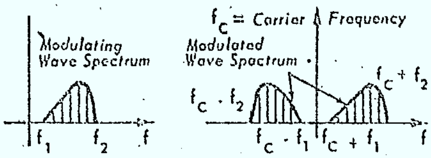


Fig. 4: This diagram shows the amplitude modulation relationship of the upper and lower sidebands and how the information can still be transmitted by eliminating one sideband by using filters.

Modulating and demodulating equipment is expensive; but multiplexing is essential to reduce the number of coaxial cables and mi-

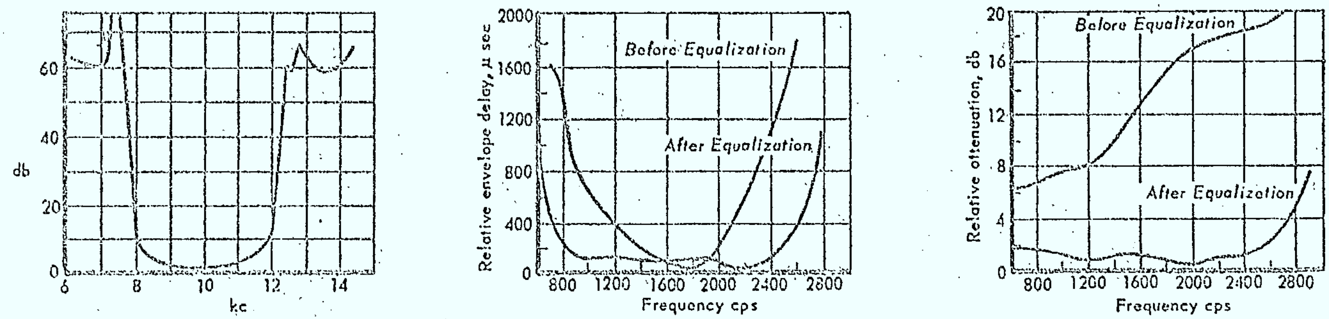


Fig. 2: Typical characteristics of voice grade channels

crowave links. These links, used at their fullest capability, can transmit up to 1860 channels by juxtaposition of 4KHz bands.

Figure 7 shows the L1 carrier system used by Bell Telephone. It has been developed for coaxial cable transmission and is also used on microwave radio systems. It is typical of several systems in use.

Note that the frequency translation is accomplished in several stages:

- 12 channels are translated to the band 60-108 KHz to form a group.
- 5 groups are translated to the band 312-552 KHz to form a supergroup.
- 10 supergroups form a master group (60 KHz to 3.1 MHz).

By means of this three-stage structure, wider bandwidths than the basic telephone channels are available; in particular the 48 KHz group band, and the supergroup band. With the advent of the picturephone, the standard 1 MHz channel, named Telpak D should also become available in the future.

Low-speed data communication is now achieved through teletypewriter channels, which utilize, at a speed of 10 five-bit characters per second a bandwidth of 170 Hz. Up to 18 such channels can be multiplexed into one single telephone channel, although for practical considerations a full channel is often used. For example, when the tele-

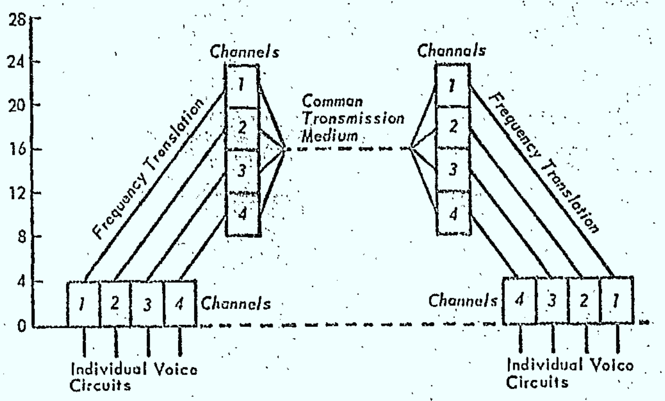


Fig. 5: By using a number of different carrier frequencies in say voice transmission, the same voice bandwidth (200 to 4000 cps) can be translated to separate portions of the frequency spectrum before being applied to a common transmission medium.

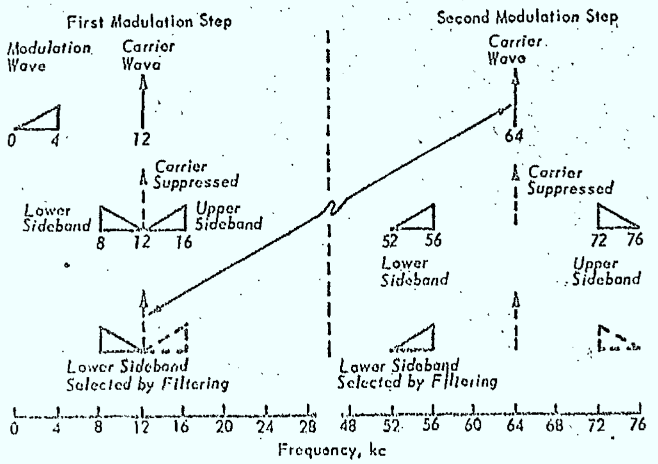
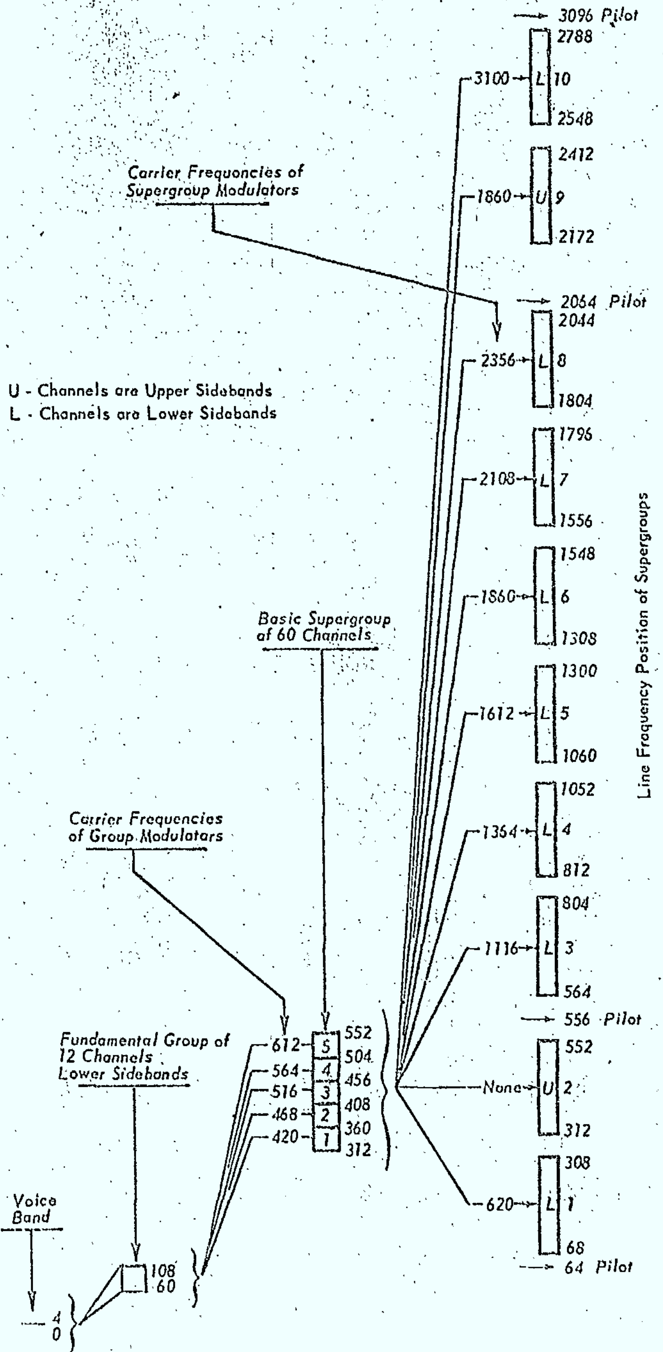


Fig. 6: In SSBSC (Single Sideband, Suppressed Carrier) modulation, the carrier and one sideband are removed in each channel used. Note at the left the transmission band will be from 8 to 12 KHz and at the right it will be 52 to 56 KHz. A great many such channels can be accommodated in the total usable frequency spectrum.



3096 Pilot
2788
3100
L 10
2548
Carrier Frequencies of Supergroup Modulators
2412
1860
U 9
2172
2064 Pilot
2044
2356
L 8
1804
U - Channels are Upper Sidebands
L - Channels are Lower Sidebands
1796
2108
L 7
1556
1548
1860
L 6
1308
Basic Supergroup of 60 Channels
1300
1612
L 5
1060
1052
1364
L 4
812
Carrier Frequencies of Group Modulators
804
1116
L 3
564
556 Pilot
Fundamental Group of 12 Channels Lower Sidebands
612
564
516
468
420
5
4
3
2
1
552
504
456
408
360
312
552
None
U 2
312
308
620
L 1
68
64 Pilot
Voice Band
4
0
108
60
Line Frequency Position of Supergroups

Fig. 7: This shows the L1 carrier system and its allocation plan used by Bell Telephone. It has been developed for coaxial cable transmission and also microwave transmission systems.

type is linked to a time-sharing computer by normal telephone line, a simple drop circuit, that is, a pair of wires, is all that is required if computer and user are in the same city. In fact in most connections of up to 40 miles, only drop lines, not a multiplexing technique, are employed.

The defects of the channel depend on its type. Noise in telephone communications comes from many sources, such as cross-talk (interference from other lines), switching noises, echo in long distance communications, etc.

Noise disturbances in all channels sometimes occur in bursts, which virtually 'black out' all transmissions, for a short interval of time. Synchronization between transmitting and receiving equipment is then lost and must be recovered quickly.

When a multiplexed channel is demodulated, the high frequency carrier is recreated. A small error in that frequency will cause a translation of the signal spectrum.

In Figure 8:

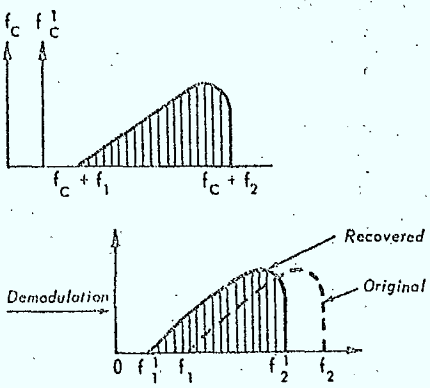


Fig. 8: This diagram shows how an error of .01% of the frequency of the reinserted carrier at the receiving can cause a significant translation of the voice band.

f_c is the original carrier frequency: (say 68 KHz).

f_1 is the lowest frequency of the voice signal spectrum (say 200 Hz).

f_2 is the highest frequency of the voice signal spectrum (say 3000 Hz).

$f_3 = f_c + f_1$: lowest frequency of the transmitted signal (68200 Hz).

$f_4 = f_c + f_2$: highest frequency of the transmitted signal (71000 Hz).

f'_c is the recreated carrier.

If an error of .01% is made in the frequency, or 7 Hz; $f'_c = 68007$ Hz.

After detection using f'_c, the voice signal will lie between: $f'_1 = f_3 - f'_c = 193$ Hz and $f'_2 = f_4 - f'_c = 2993$ Hz.

All frequencies have been shifted by a significant amount, even though the relative error in recreating f_c was extremely small.

Care must be taken in transmitting data through multiplexed channels affected by such frequency shifts.

As far as the wideband data transmission channels are concerned, AT&T has tariffed them as TELPAK channels. In such service, the bandwidths are:

TELPAK A, 48 KHz (1 group)
TELPAK B, 96 KHz (2 group)
TELPAK C, 240 KHz (1 supergroup)
TELPAK D, 1 MHz identical to picturephone channels.

The Western Union Telegraph Company also provides an 8 MHz channel (for missile-tracking systems), using a radio relay system.

As further data transmission techniques are developed, other bandwidths will likely be offered by the common carriers in the future. Picturephone will almost certainly be the most significant device implemented within the foreseeable future.

Principles of data communications

This section deals with the problem of sending data through a telephone channel. Modems (sometimes referred to as data-sets) are designed to match the voice grade channel.

Our goal is defined as follows: given a stream of m binary pulses, arrange a signal to be transmitted through a noisy channel and recover at the other end the original m pulses with an arbitrarily set maximum error rate.

In the next paragraph, I shall always be referring to the BASEBAND. Earlier, it was seen that the modulation of a band, say of 200 to 4000 Hz by a carrier at 60 KHz, gives two side bands, symmetrical with respect to 60 KHz. The Lower Side Band (LSB) extends from 56 to 59.8 KHz and the Upper Side Band (USB) from 60.2 to 64 KHz. This amounts to a simple translation of the frequency spectrum of the band. Similarly, since a telephone line is Band Pass in nature, we can define a Baseband from 0 to a certain frequency (for instance 600 Hz), and then modulate it with a carrier at the center of the voice band (say 1800 Hz) for transmission through the channel.

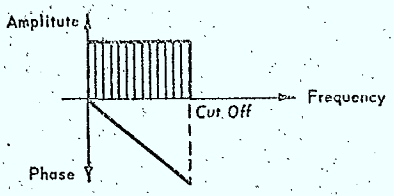


Fig. 9: This diagrammatically represents an ideal low-pass filter.

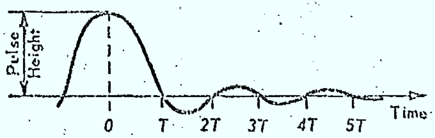


Fig. 12: Sinc Function: Impulse response of an ideal low-pass filter.

To consider this example, let us assume that we have an ideal low-pass filter (Baseband filter) as in Figure 9. If we send a pulse through it, the output is the familiar sinc function (see Figure 12). This function has the particularity of being zero at all times (k/2T), except for k = 0, T being a period defined by the baseband filter.

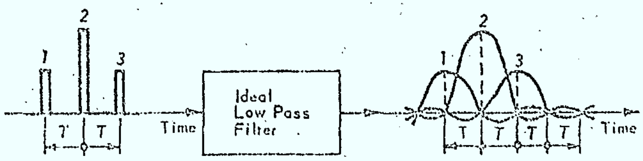


Fig. 11: Signalling at speed 1/T pulses/sec through an ideal low-pass filter shows how the response of the filter to pulses 1, 2 and 3 are the sinc function curves 1, 2, and 3 at the right showing zero intersymbol interference.

In our example:

Cut-off frequency of the baseband filter = 600 Hz
$T = 1/600$ sec.
Sinc function zero every $T/2 = 1/1200$ sec.

It is now apparent that we can conceivably send a pulse through that filter every 1/1200 sec. without Intersymbol Interference (in other words, at that rate of speed, the output of any single pulse will not interfere with the output of all other pulses). An illustration of that fundamental concept is shown in Figure 11.

A Baud is a time interval during which a unit of information is sent. One pulse per baud is transmitted: the Baud rate in our example is 1200, corresponding to 1200 pulses per second.

From this point the Bit rate (or number of binary digits of information) is easily defined. The height of the output wave at T out = 0 (see Figure 12) is directly proportional to the strength of the pulse — or energy — applied at the input of the filter.

If we recognize two independent equally probable levels (for instance: pulse or no-pulse) depending on the height detected at the output, we have one bit per signal; it is then said that the bit rate is equal to the Baud rate (see Figure 13). If we differentiate between four —all independent and equally probable — amplitude levels for one pulse, 2 bits of information can be contained in that pulse (see Figure 14). This creates a bit rate of 2400 for a Baud rate of 1200. In a telephone line, the use of a 2400 Hz bandwidth means that a Baud rate of 4800 can be achieved with 2-level logic. Using a 4-level logic (2 bits per pulse), a bit rate of 9600 is possible.

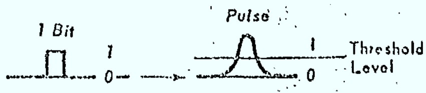


Fig. 13: Two-level detection: One level of detection is pulse or no pulse. The second level is if the output is below a given threshold level — read 0. If the output is above the threshold level — read 1.

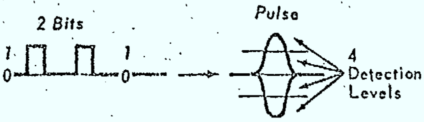


Fig. 14: Four level detection: Each detection level corresponds to one of the four possible combinations of two bits of information. Bits are transmitted in four combinations — 00, 01, 10, or 11. These four conditions can be coded into pulses that effectively double the bit rate for a given line speed.

Pulse Width, as opposed to pulse height can be preserved by means of a cosine roll-off baseband filter with linear phase (see Figures 15 and 16). The same basic concept applies.

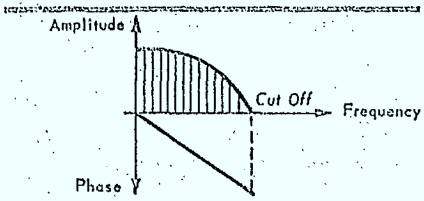


Fig. 15: Cosine Roll-off Filter: Preservation of pulse width is achieved by using this baseband filter.

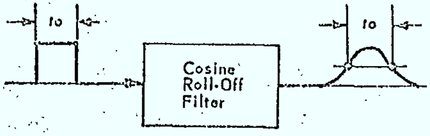


Fig. 16: Pulse width preservation

Nyquist originally derived those results. Figure 9 shows the ideal baseband filter demonstrating the Nyquist criterion No. 1 or pulse height preservation; and Figure 15 shows the ideal baseband filter demonstrating the Nyquist criterion No. 2 or pulse width preservation. Note that real filters — see Figure 17 for a real filter approximating the ideal low pass of Figure 9 — allow the realization of no-symbol interference in an extension of this theory, achieved in practice in the 4800 bits/sec ESE modem.

This signalling theory applies only to an ideal channel, free from noise and distortion. The effect of phase (and also amplitude) distortion is to change the shape of the output pulse so that the zerocrossings (see Figure 12) will not occur at the expected times and will create intersymbol interference.

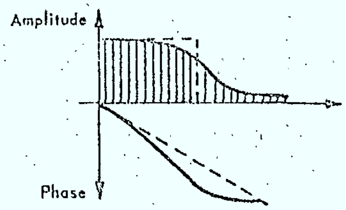


Fig. 17: Approximation to an ideal low pass filter

Intersymbol interference is not encountered when a narrow band at the center of the telephone line is being used (i.e. for low speed data transmission); but equalization — or line conditioning — of the line becomes necessary for high speeds. Hence the adaptive modem: at high speeds (say 9600 bits/sec) an adjustable — sometimes automatic — equalizer is part of the receiver, to compensate for channel distortion. If the communication is to be done on the switched network, where the characteristics of the channel being used are not known in advance, the equalizer needs to be adjustable.

Another crucial factor is the amount of noise in the channel. The signal to noise ratio of the signal power over the power of the noise must be known. With perturbed output, there is uncertainty in recovering the original information. The maximum possible capacity is then:

$$C(\text{bits/sec}) = W \log_2 \frac{S+N}{N} =$$

$$W \log_2 \frac{S}{(N+1)}$$

W: bandwidth of the channel (in Hz)
S: average signal power
N: noise power

This formula states that if N = 0 (noiseless channel) the information rate is theoretically unlimited, i.e. at every pulse an infinite number of pulse amplitudes or pulse widths (or pulse phases in different modulation schemes) are recognizable, hence an infinite amount of information is carried. With a Signal to

Noise power ratio of 31 and a Channel Bandwidth of 3500 Hz:

$$C = 3500 \log_2 \frac{31+1}{1} = 3500 \text{ x } 5 = 17500 \text{ bits/sec.}$$

The price of a modem will necessarily depend on its complexity. Cost and channel distortion considerations will normally limit the bit rate to a somewhat lower figure. Presumably the advent of Large Scale Integration of digital circuits will allow the engineering of more complex networks for a reasonable price and we are looking forward to a wider range of speeds available on a telephone channel.

The Amplitude Modulation scheme of the common carrier structure outlined earlier allows us to develop the Nyquist theory using the Baseband only because a frequency translation is employed. There are, however, many other modulation schemes, which do not involve amplitude modulation. The frequency of the carrier can be changed according to the signal (FM); (see Figure 18) the Phase of the carrier can be controlled to represent a coded information (PSK or Phase Shift Keying) — (see Figure 19). Such schemes do not lend themselves to baseband analysis; a direct analysis (sometimes quite difficult) must be done.

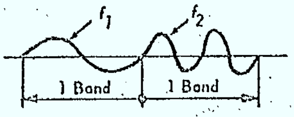


Fig. 18: Frequency modulation. Instead of transmitting actual pulses over the medium, a given carrier frequency can instead be shifted for the duration of the pulse to a higher or lower frequency. This shift and its duration can be detected and converted back to a shaped pulse. If f_1 is the carrier it could be read as no pulse present and the bit value would be 0. If f_2 represents the frequency produced with a signal present its bit value would be 1.

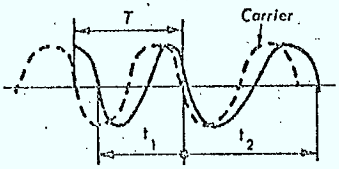


Fig. 19: Phase shift keying (PSK). The phase of a sine wave signal can be advanced or retarded relative to an original reference point through the use of phase shift networks. If t_1 is [illegible] than T — the bit value may be 1 (or 0); if t_1 is greater than T — the bit value may be 0 (or 1).

Given a fixed bandwidth and a fixed signal-to-noise ratio, the channel capacity employing any one of these schemes remains the same. The choice between these Modulation schemes depends on such criteria as:

- ○ Simplicity of the circuits.
- ○ Convenience of use (ease of synchronization with the equipment sending and using the data).
- ○ Resistance of each of the schemes to particular types of channel distortions.

Most of these matters are the object of active research at the present time.

Techniques used in data communications

The most widely used techniques used in the Data Communications industry are as follows:

MODEMS

Modulation Schemes

○ **Amplitude Modulation:** SSB (Single Side Band) is used very widely because it makes efficient use of its assigned bandwidth.

○ Its implementation in modems is the VSB scheme (Vestigial Side Band).

Frequency Modulation (FM) and **Phase Modulation** (PM) are attractive when simplicity of equipment is required. FM is invaluable for low speed data transmission, because the bandwidth it requires can easily be provided. It resists some channel imperfection such as amplitude variation. Teletype systems often use FM modulation.

Phase-Shift Keying (PSK) is mathematically identical to FM. However, the distinction lies in the difference between the equipment required to detect frequency and phase variations. PSK has been employed in high-frequency applications where fading is a problem. The receiver derives its information from the phase difference between two successive pulses. If the perturbation is slow compared to the intervals between pulses, successive pulses will be disturbed by the same amount and the information will be required successfully.

Quadrature Amplitude Modulation (QAM), a scheme recently implemented by ESE Ltd., is theoretically as efficient as VSB.

The message is divided into two halves and each half is used to generate a different baseband signal. The first baseband modulates the carrier and the second one modulates the carrier in quadrature with the first one.

The resulting signal is a combination of the two basebands, and has theoretically the same bandwidth and the same information content as a signal obtained with a VSB scheme.

Types of operation

At low speeds (up to 1800 bits/sec) the operation of the modem can be asynchronous. The transmission speed is allowed to vary within certain limits. At speeds at and above 1800 bits/sec the receiver usually expects data at a fixed rate; synchronous operation of the modem requires joint timing of transmitter and receiver.

Line Conditioning must be used for non-adaptive circuits in high speed applications. At the present time, dedicated data circuits are established, which can be compensated once and for all by equalizer networks.

An adaptive modem is often required on the switched network and as its price goes down with the advent of more efficient techniques and cheaper integrated circuits, these modems will be widely used. The trade-off between speed and price is dramatized here:

A 1200 bit/sec modem costs about $800.

A 2400 bit/sec modem costs about $2200.

A 4800 bit/sec non adaptive modem costs about $5000.

A 4800 bit/sec adaptive modem costs about $12000.

COMMON CARRIER

Multiplexing methods

Frequency Division Multiplexing can be composed of frequency bands, coming from Amplitude and/or Frequency modulated carriers. In FM, the frequency spectrum being used is a function of an arbitrary modulation index; the frequency deviation is proportional to the amplitude of the signal. The band occupancy of an FM signal is a function of that index. Indeed, this is the basic advantage of FM: the ability to exchange bandwidth occupancy for noise performance (as Bandwidth increases, so does noise immunity) by varing this index.

Time Division Multiplexing: TDM works on the basis of allocating 'time slots' to each channel rather than whole frequency bands. Figure 20 illustrates the concept of TDM.

A sine wave must be sampled at least twice during each cycle in order to reconstruct it accurately at the receiver. If the highest frequency to be transmitted is 4 KHz, the minimum sampling rate is 8000 samples per second, or one every 125usec. According to the length of each transmitted pulse, the space between pulses can be used for other channels.

In theory the number of channels possible per cable is equal whether using Frequency Division Multiplexing or Time Division Multiplexing; but the timing equipment required for TDM is very critical.

The three modulation schemes commonly employed in TDM are:

PAM (Pulse Amplitude Modulation).

PDM (Pulse Duration Modulation).

PPM (Pulse Position Modulation). Figure 21 shows the principle of these schemes.

PCM — Pulse Code Modulation — is a 2-level PAM scheme: discrimination at the receiver is performed between pulse and no-pulse. For instance, if the message is an analog signal, the sampled values are coded into an arrangement of several binary digits (0 for no-pulse and a fixed amplitude for pulse). m pulses will represent 2^m different finite levels in the original signal. Quantizing must be accomplished as shown in Figure 22. This PCM is efficient in overcoming noise and interference, as degradation of the shape of a pulse will not affect the message. PCM schemes are employed in transatlantic cables.

THE CONCENTRATOR

As the load increases on the communications network, research installations are increasingly probing the problems of efficient pathfinding and full usage.

Data communications are being developed to the point where efficient use of existing lines can be made. This is largely due to the current development of the concentrator.

The concentrator is a small special purpose computer that will send and receive data to and from several users at the same time using the same line, allocating 'time slots' to each user according to the amount of information that is sent or received. Buffers will store some messages for a short time if many users at one particular time want to send data.

Similarly, in a transatlantic cable, two pairs of wires are required per conversation, even though, on the average each circuit is used at 50 percent efficiency in each direction. A concentrator, using the lines at 100 percent efficiency in each direction will permit twice as many conversations with the same number of wires.

It is apparent that, as in the implementation of Multiplexing, the extra equipment required at each end of the line will pay for itself many times over.

Conclusion

It is hoped that the success of the new communication devices, such as adaptive modems and concentrators, in cutting costs and improving efficiency in the data communications field will have a far reaching effect. In spite of inflation, we may well see, someday, two bits worth less than a quarter. ◙

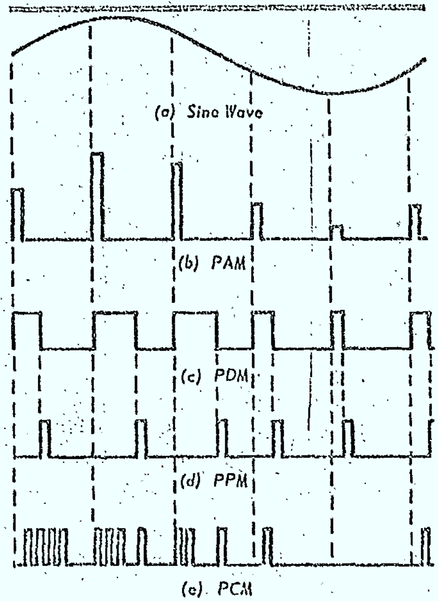


Fig. 21: This illustrates the kind of pulses transmitted in time division multiplexing using various modulation schemes. If (a) is the sine wave to be transmitted, (b) shows what PAM (pulse amplitude modulation) would appear like; (c) PDM (pulse duration modulation); (d) PPM (pulse position modulation) and (e) PCM (pulse code modulation).

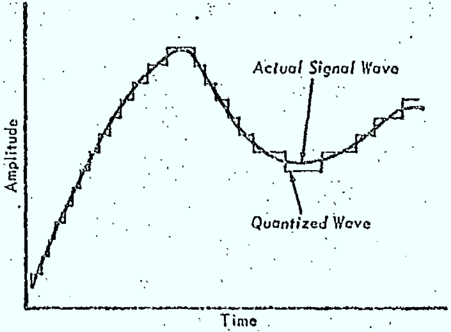


Fig. 22: Quantizing of the original signal for PCM is accomplished as shown. The analog signal is sampled and coded into an arrangement of binary digits to indicate the amplitude level. This transmission system is efficient in overcoming noise and is used in trans-oceanic cable systems.

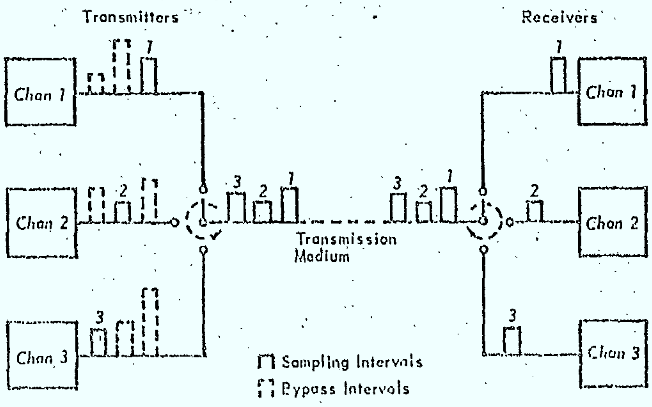


Fig. 20: In time division multiplexing, a number of circuits share the same transmission medium but at different times.

Data transmission: a direction for future development

Because the public telephone network was designed only for voice communication, problems arise when data communication is attempted on this same network. Data waveforms can be converted for ease of handling but it is difficult to match the discontinuous flow of data from a terminal to the continuous flow of information in the communication channel

Harry Rudin, Jr. ***IBM Zurich Research Laboratory***

Despite the rapid advances in many regions of data transmission, there is a rapidly growing number of applications for which existing data-transmission techniques are inefficient. A look at the status of data-transmission development indicates that, although a very successful campaign has been waged to map the data waveform into a waveform ideally suited for transmission on the communication channel, very little has been done to match the often discontinuous flow of data from the terminal to the continuous flow of information in the channel. Combining randomly occurring messages from several sources into a more continuous flow is described by the mathematics of traffic theory. Although this theory has been extensively applied to speech traffic, it is rarely applied to data traffic. As a specific example of the gains in channel efficiency that can be had through the application of traffic theory, the multiplexer–concentrator is examined. In the author's opinion, it is in this area of application of traffic theory to data communication that many of the more significant future developments in data transmission will be made. To be sure, some work has been done, but it is relatively little when the gains that can be made are considered.

The public telephone network, naturally enough, was designed only with voice communication in mind. The result of this design is that efficient data communication over the telephone network is difficult to achieve because a number of factors that do not affect the quality of voice transmission seriously impede the flow of data. Delay distortion in the frequency domain is an example of one of these phenomenons.

A tremendous research and development effort has been made to overcome many of the restrictions imposed by the voice-communication network. The bit rate (and the reliability) at which data can be transmitted has been markedly increased within the last few years as the result of such technological advances. But the strange and fascinating aspect of this effort is that it has been carried out almost exclusively in the domain of signal design. That is, research has been almost exclusively concentrated on the development of techniques that convert the data waveform into another waveform that will easily pass through the transmission channel.

However, the public telephone network was designed not only with the voice waveform in mind, but with the *statistics* of voice communication in mind as well. Although it has been well-recognized that the telephone network is not suitable for transmission of many pure data waveforms, the assumption is usually made (albeit tacitly) that the statistics of data communication allow direct transmission of data over the telephone network. This—more often than not—is a bad assumption.

Traffic theory is the study of the statistical flow of messages and this theory is largely responsible for economical voice transmission as it exists today. In fact, the development of the theory was motivated mainly by telephony itself. Unfortunately, traffic theory is only now beginning to be applied to the data-communication problem.

In this article, the present direction of data-communication development and possibility for improvement is assessed in a cursory fashion. Next, the efficiency of conversational and interactive data-communication systems is examined. Finally, an example (slightly rigged to be sure) is considered to illustrate the gains that can be made in certain circumstances. The point of the example is to show that there are cases in which tremendous increases in channel efficiency can be made by taking traffic statistics into account.

Recent developments in data transmission

Recent developments in data transmission can be divided into three areas: modulation techniques, equalization techniques, and error control.

The term "modulation techniques" is used in the most general sense—i.e., the modulation process is that process which maps the data signal (often digital) into a form that is suitable for transmission over whatever transmission facility is at hand. It might include serial-to-parallel conversion, translation from baseband to a higher frequency, and band-limiting methods. There are two techniques very much in vogue.

The first modulation technique is multilevel baseband transmission combined with vestigial sideband modulation.[1,2] Multilevel transmission permits several bits to be sent at the same time so that higher speeds may be achieved. The vestigial sideband modulation is a good, practical approximation to achieve the spectrum utilization efficiency of single-sideband modulation.

A second technique is that of partial response.[3] In this technique a correlation is introduced between what is transmitted at one signaling instant and one or more other signaling instants, with the result that various spectral shapes can be obtained. Practically, the advantage of this family of techniques is that it permits transmission at new combinations of speed and sensitivity to noise.[4]

The developments in modulation techniques have permitted very efficient design of modems that convert the data signal into a form ideally suited to a specific, known communication channel (the average telephone channel, for example). Unfortunately, the difference between a specific channel and the average channel can be quite large, and this difference (in addition to manufacturing tolerances) was the major hindrance to efficient data communication until a few years ago. The development of a variety of automatic equalization techniques has since removed that hindrance.[5-7] Using these techniques, it is possible to compensate automatically for the variations of the individual channel and also the manufacturing tolerances.

One of the more up-to-date devices for this application is the transversal filter. It provides a dynamic and flexible approach to the problem of equalization and is well-suited to adaptive as well as automatic operation, qualifying it for use when distortion is time variable.

A very large percentage of the literature devoted to communications and information theory concerns various coding and decoding techniques for the detection and correction of errors incurred in the process of data transmission. It is possible to design a data-transmission system that operates with a very high speed but with an unacceptably high error rate.[8] However, by using a small percentage of the transmitted information for the insertion of redundancy, a disproportionately large gain can be made in the reduction of the error rate. Unfortunately, the coder and, particularly, the decoder in the vast majority of cases described in the literature are prohibitively expensive. However, these techniques can be both practical and economically reasonable.[9]

The techniques just described have been put into practice in the case of the telephone channel, and the results are truly impressive. For example, it is possible to transmit at the rate of 9600 bits per second (b/s) over carefully prepared leased lines.[10] Also, very extensive

FIGURE 1. Simplified data-communication complex.

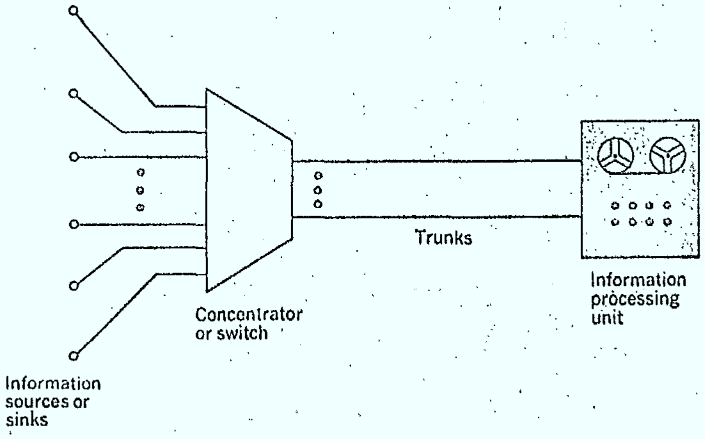
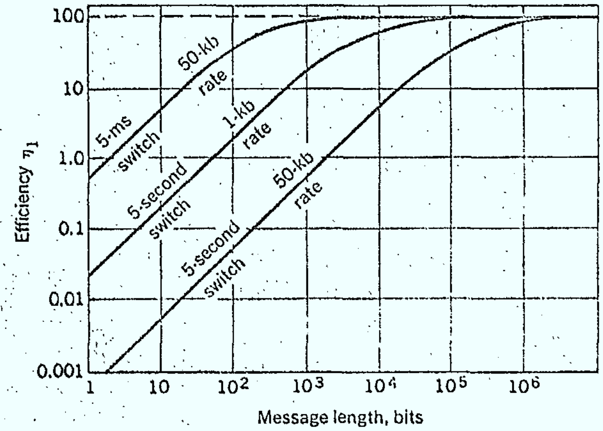


FIGURE 2. Efficiency of some data-communication systems—dial and transmit (computer processing included).



tests have been made over the Direct-Distance Dialed (DDD) network to prove the reliability of 3600-b/s transmission over unprepared facilities.[11] Experimental work indicates the possibility of still higher speeds over the DDD network. Unfortunately, these units are very complex and the complexity seems to increase rapidly with speed.

The question should be asked, "How much room for improvement is there?" Shannon's theory[12] provides the answer for error-free transmission in a white-noise-only environment in terms of the following equation:

$$C = W \log_2 (P/N + 1)$$

where W is the channel bandwidth in hertz and P/N is the signal-to-noise power ratio. Estimating the bandwidth of the voice channel to be 2400 Hz and the signal-to-noise ratio to be 30 dB results in a very rough approximation of the channel capacity: C equals 24 000 b/s. There is, then, room for improvement, but not very much. What progress there is to be made in this direction is apt to come at great cost. This is not to say that the advent of large-scale integration (and with it the promise of economical digital- and active-filter technologies, for example) will not sharply reduce the cost of modems even more complex than the ones produced today. However, there may well be a more profitable direction in which to carry out research.

In summary, much effort has been expended and much success has been had in matching the transmitted signal waveshape to the channel's characteristics. In the voice-communication-channel area, excellent work has been done in making the data waveform look like the speech waveform so that it will easily pass through the channel designed for speech.

Data-transmission system efficiency

As long as the flow of data is continuous, the conventional data-transmission system performs very well indeed. But when the flow of data becomes nonregular or random, the data-communication system can be rather inefficient. The vast majority of research and development efforts in data transmission have been devoted exclusively to the case of continuous or nearly continuous data flow. A look at one of several books devoted to data transmission will verify this.[13,14]

Unfortunately, there is a rapidly growing number of applications in which the flow of data is far from continuous. One example is found in a large inquiry system. Here, several remotely located or satellite terminals make requests to a central computer—perhaps to ascertain a customer's credit, to find out if a seat is available on an aircraft flight, or to check the inventory of a certain item. Both the request for information and the required information are usually very short messages; further, these short messages are very likely separated by long idle-time periods.

Another example of discontinuous data flow is the use of a time-shared computer, perhaps for scientific programming. In this case, the likelihood is that a communication channel will remain idle for long periods of time while the programmer or the computer ponders some aspect of a problem.

One way of characterizing the continuity of data communication is by means of message length. Given the message length, the bit rate of the communication channel, and the time required to set up the connection, it is possible to determine the efficiency of a communication system in terms of the percentage of time actually spent transmitting needed information. The efficiency η_1 is given by

$$\eta_1 = \frac{\text{Time required to transmit the information}}{\begin{array}{c}\text{Total time required to make the connection}\\ \text{and transmit the required information}\end{array}}$$

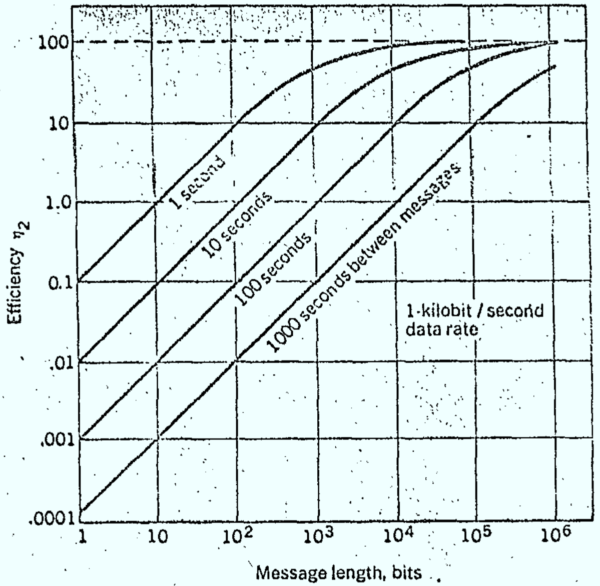
For the very simple, single-switch communication system shown in Fig. 1, η_1 can be plotted as a function of the message length for assumed switching time and rate of transmission as shown in Fig. 2. Two data rates are assumed: The 1-kilobit-per-second (kb/s) rate is taken as a fairly representative number for many data modems capable of operating on the Direct-Distance Dialed telephone network; the 50-kb/s rate is taken as representative of a wide-band data service or of a single pulse-code-modulation voice channel.

Two switching or interconnecting times are assumed: 5 seconds to represent a conventional, mechanical switching system and 5 ms to represent a futuristic solid-state switching system.[15] (It should be noted that these switching times include the processing time of a computer used to control the switch itself.) Combinations of these parameters and a specified message length determine the efficiency.* The results of this simple calculation are shown in Fig. 2. It is clear that in many cases of interest the efficiency is indeed low. Of course, this is not the only criterion of goodness that exists for the evaluation of a communication system.

In the face of such low efficiency, there is another path open to the user. Rather than reestablish the connection for each call (and thereby suffer the penalty of connect time), the user can establish the call only once and ac-

* The combination of 1-kb/s transmission speed and 5-ms switching time does not appear as it is an unreasonable combination. In the 5-ms period there is insufficient time to transfer the information necessary to establish the call, even at the 1-kilobit rate.

FIGURE 3. Efficiency of some data-communication systems—hold and transmit—at a 1-kb/s data rate.



cept, instead, the penalty of holding time between messages when the line is idle. A simple calculation can also be made for this case where the efficiency η_2 is now defined as follows:

$$\eta_2 = \frac{\text{Time required to transmit the information}}{\text{Sum of average holding time between transmissions and time required to transmit information}}$$

The efficiency η_2 is plotted as a function of the average message length for a number of assumed holding times between transmissions in Fig. 3. A data-transmission rate of 1 kb/s is assumed; another set of curves could be drawn for transmission at 50 kb/s with a corresponding degradation in efficiency. Again it is seen that the efficiencies are not high for many cases of interest.

The recognition of such inefficiencies is not new. In a forward-looking paper[16] published in 1961, the authors calculate a "system utilization index" or efficiency for a specific system of some 2 percent. Unfortunately, not a great deal has happened in the intervening period. For example, a very recent (and very timely) paper by Jackson and Stubbs gives a range of 1 to 5 percent for the communication efficiency of several multiaccess computer systems.[17]

So far, only the point of view of the communication user has been considered, but the present state of affairs is also a source of difficulty to the communication supplier. If the customer chooses to operate in the first mode (dial and transmit) and makes many such transactions, the computer that directs the switching in a modern exchange may become overloaded. If the customer chooses to operate in the second mode (hold and transmit) many trunks are blocked for very long periods of time. Thus there is pressure for change from the communication supplier as well.[18]

These inefficiencies are all a result of the nonuniform character of data flow or, in other words, the traffic statistics of data. The job of converting the often discontinuous data waveform into a continuous waveform suited to the transmission channel's capabilities has been well done. The job of converting the often discontinuous traffic pattern of data into the continuous flow of information in the transmission channel has barely been begun.

Some approaches to a solution

In order to discuss and compare several solutions for the problem of data communication with random-message flow, these solutions will be discussed with respect to a specific system—that shown in Fig. 1. The geographical distribution implied in Fig. 1 (a cluster of terminals far removed from a common destination) is chosen to emphasize the gains in efficiency that can be made if circumstances are ideal.

Although the geographical clustering of terminals in Fig. 1 may seem arbitrary, such clustering is commonly observed.[19] Yet another system might be one wherein communication costs are negligible and no extra hardware costs are warranted to reduce already negligible communication costs.

Obviously, what is needed to improve trunk-line efficiency is a technique for combining the communication needs of many terminals (when this is feasible) so that the total data-communication flow becomes more regular and the communication capacity of the trunks better utilized. In fact, this approach has been taken in some nonvoice transmission cases, but infrequently and for relatively long messages—telegraph messages, for example.[20]

The most straightforward approach is the use of a pure multiplexing technique. Multiplexing is here defined as a technique that assigns portions of the channel's capacity to various users on a fixed, a priori basis. A multiplexer has the same total input and output capacities determined on an instantaneous basis.

The channel capacity can be subdivided either in frequency or in time. Compared with transmission on an entire trunk, each terminal "sees" a reduced capacity. When this reduced capacity can be tolerated (for example, when possible resultant delays are reasonable), the restriction on transmission rate itself acts as a filter to smooth the varying traffic flow. Since several terminals can use the same trunk, utilization or efficiency is higher.

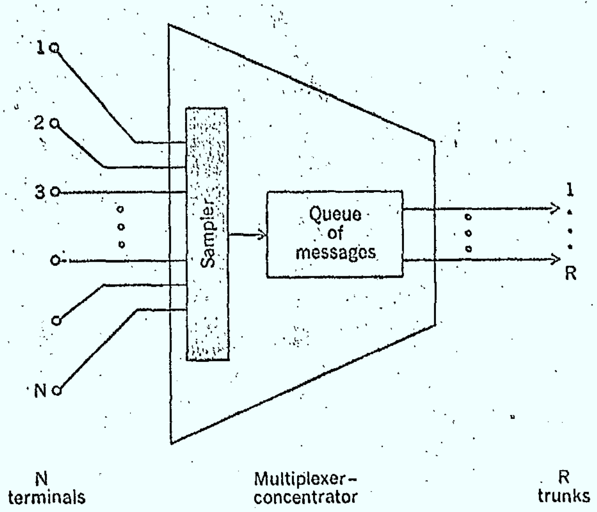
Frequency-division multiplex (FDM) is a well-used technique in the telephone system.[21] In FDM, a segment of the channel's frequency range is allocated to each subchannel. The other multiplex possibility is time-division multiplex (TDM). This is the approach used for subdividing pulse-code modulation channels[22] and in which time slots rather than frequency slots are assigned to subchannels or terminals.

FDMs and TDMs are made by a number of manufacturers for subdividing the voice channel into a number of low-speed data channels.

A more flexible system can be had by using concentrating, as opposed to multiplexing, equipment. "Concentration" is here defined as the assigning of the trunk's capacity on a dynamic or demand basis. A concentrator, however, may have more input that output capacity and so may generate errors.

A concentrator can be simply a relatively fast switch that connects the various terminals, upon request, to the lines or trunks leading to the information-processing unit. The switch can be made to function very rapidly because it has a very limited number of customers and need only perform a limited number of tasks for these

FIGURE 4. Message-multiplexer system.



customers—in contrast to a conventional telephone exchange.

If two messages require transmission at the same time in this approach, one must wait until the other's transmission has been completed. One such system has been described by J. J. Watson.[23] The Bell System has recently announced DATREX,[18] which performs a similar function. Devices such as these are essentially line-switching concentrators.

Although the line-switching concentrator permits significant improvement in the utilization of transmission facilities, it does have its limitations. This is so because it is restricted to setting up a line connection and maintaining that connection until the entire message, including the possibility of some idle time, has been transmitted. Other messages cannot be accepted by the concentrator switch until the first message has completely cleared the system. The signaling techniques (used for specifying the connection path and end of message) are conventional, thus limiting the connection speed.

A system that is capable of higher transmission efficiencies is the multiplexer–concentrator with buffer storage, shown in Fig. 4. This system has three main features:

1. Signaling (consisting only of the address of the calling terminal) is handled by the multiplexer–concentrator and at the trunk's bit rate.

2. The messages are of fixed length but can be very short—perhaps only a single bit or character—and each message is addressed.

3. A queue or waiting line for messages is provided so that every terminal may make the assumption that the system always has room for another message.

As a result of these features it is clear that this system also has several disadvantages.

First, the maximum efficiency of the system is limited—a result of the fact that the address is transmitted with each message. If, for example, five bits are used to represent the message (a single character) and there are $32 = 2^5$ terminals, the maximum attainable efficiency is 50 percent (five information and five address bits). In contrast, the line-switching concentrator may approach 100 percent, but only for very long messages. A glance at either Fig. 2 or 3 reveals, however, that 50 percent or even 17 percent (if the message is a single bit in the example above) is a relatively high efficiency compared with that of many conventional systems.

Second, the system is not a very flexible system because of the heavy dependence of efficiency on the number of addresses.

Third, since the queue must be of limited size in a realizable system, there is a definite probability of error in this system. This is a key parameter and will be discussed further.

The notion of such a multiplexer–concentrator is not new.[16] In the field of digital computer input–output systems, a particularly large effort has been made in this area. In fact, such an approach is mandatory if a computer memory is to have the capability of communicating efficiently with card readers, punches, printers, disks, tapes, and terminals.[24,25] The preoccupation of the designers of these internal multiplexers for large digital computers has been to provide the most efficient use of the computer's memory in an effort to minimize the interference of the communication process with the internal operation of the computer. What would seem desirable in the case of the time-shared computer system is the extension of the multiplexing from inside the computer to a point closer to the remote terminals so that the communications facilities can be efficiently shared as well as the computer's memory is.

Some work has been done in this direction by essentially using a full-size digital computer to achieve multiplexer–concentrator performance as well as to accomplish a great number of other services simultaneously.[26,27] Some computer manufacturers and some manufacturers of peripheral computer equipment do provide remote multiplexer–concentrators that are employed to increase communication efficiency. (Unfortunately, the design of such units is apparently considered highly proprietary and very little information appears in the open literature.) IBM, for example, has models 3967 and 3968 in its product line. These units, as well as the IBM model 2905 (which has recently been described in the literature[28]), can serve as multiplexer–concentrators.

A closer look at a multiplexer–concentrator

A specific form of the multiplexer–concentrator with buffer storage provides an interesting model for closer examination and demonstration of increased communication channel efficiency. The message length is chosen as short as possible, i.e., a single bit or a single character. This author has recently been engaged in a study of the statistical behavior of such a model.[29]

This multiplexer–concentrator is an interesting model because it represents an extreme in simplicity and attainable efficiency. The system examined here can be thought of as perhaps the most primitive message switch[20] because the messages are all of the same and shortest possible length. Also, the "header" (which identifies the message) has the shortest possible length, consisting only of the address. Examination of this system, then, should provide a kind of lower bound that can be improved upon by more sophisticated systems of the future.

Although the multiplexer discussed here provides an interesting model for study, it also has a number of practical advantages. One advantage is that the multiplexer looks like a variable-speed terminal to the user: Whenever the user wants to deliver a bit or a character, he may. This is important, for example, in the case of remote display of computer graphics where the need is for a burst of high-speed data followed by a long quiescent period.[30] Another advantage is that, in addition to shared costs for expensive trunks and modems, a very powerful error-control unit might be shared as well.

A high attainable line utilization or efficiency results from the statistical averaging process of the many randomly operating input terminals. Specifically, it is the inclusion of memory that allows the messages to collect randomly in a queue for subsequent transmission at a regular rate on the trunks.

As stated earlier, it is necessary to limit the size of this queue in any physical device and it is clear that such a restriction introduces the probability of error. A question that must be answered immediately is, "How large must the memory be to achieve a high information-transmission efficiency and at the same time maintain a low probability of error from overflow of the queue?" If the memory must be extremely large, the system is unreasonable from the economic point of view.

In the work mentioned earlier* such an analysis was made assuming that:

1. The messages to be transmitted are of constant length in agreement with what has been stated here.
2. The number of terminals connected is finite (specifically, 15 in the curves shown here).
3. The maximum queue length is specified.
4. The number of messages arriving during the transmission period of one message is binomially distributed; the terminals operate independently.

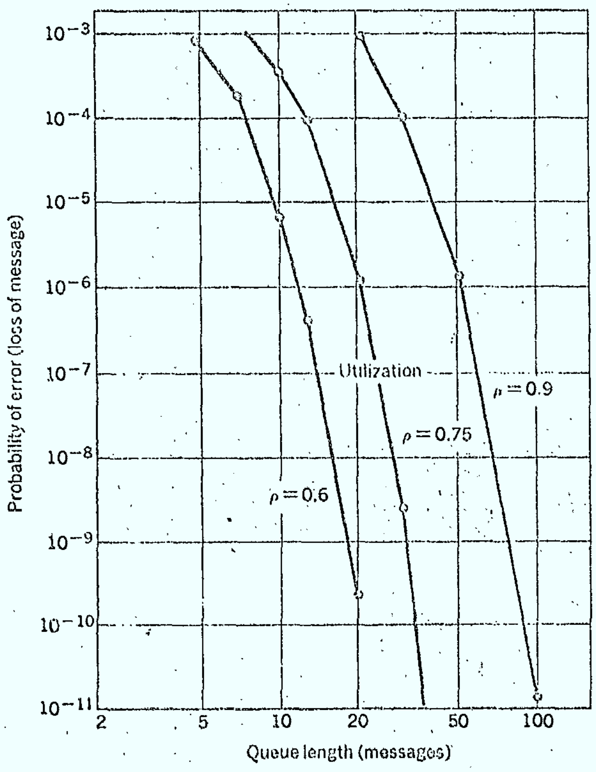
The results of this analysis are most encouraging and are shown for a single trunk in Fig. 5. The probability of message loss or probability of error is plotted as a function of the maximum allowable queue length. Three curves are drawn for varying levels of channel utilization ρ.

The channel utilization ρ is simply the percentage of the time during which the channel is carrying data. The overall efficiency of the system is thus determined by multiplying the channel utilization by the percentage of useful information in the data. For example, for $\rho = 0.75$ and the five information- and five address-bit message mentioned earlier, the overall efficiency is

$$\eta = (0.75)(0.5) = 37.5\%$$

The memory capacities indicated by Fig. 5 are quite reasonable in the light of what is expected by contemporary data-transmission standards that often have an error rate of one in 10^5. The error rate in the multiplexer–concentrator, for example, could be held to one in 10^8 with a maximum queue length of about 70 messages even for the very high trunk utilization of 90 percent.

FIGURE 5. Probability of error vs. queue length for 15 independently operated input terminals and three utilization factors.



In addition to providing the desired smoothing of data flow, the memory must introduce delay. It is, of course, important to make certain that this delay remains within reasonable bounds. Figure 6 shows the average waiting delay (in relation to the unit time required to send a single message) as a function of the maximum allowable queue size. As in Fig. 5, three values of utilization are indicated. It is evident from Fig. 6 that, even in the case of high utilization, the delays are reasonable.

As a summary, consider a system consisting of 15 high-speed (1000-b/s) terminals clustered together and connected by 2000-bit trunks to a distant information processing unit. The terminals can operate at 1000 b/s but do so only 6 percent of the time (producing uniformly distributed, single-character messages) in an independent fashion.

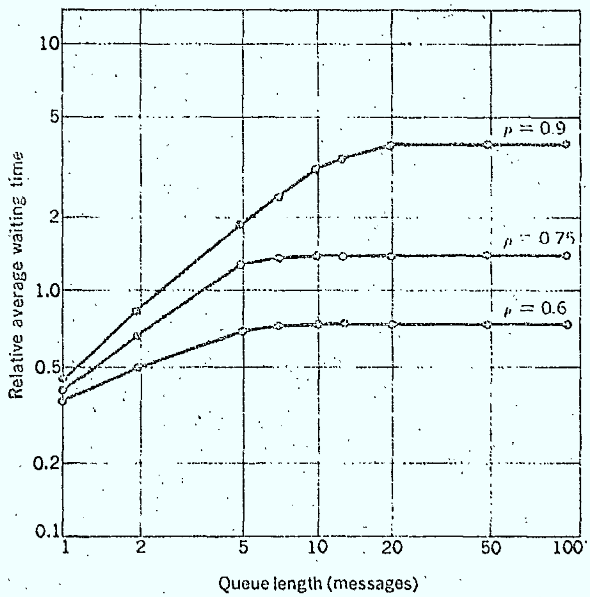
One communication possibility is 15 separate lines with an assumed error rate on the trunk of one in 10^5 messages. Alternatively, two channels can be multiplexed on each of seven channels, reducing the required number of trunks to eight. Through the use of a multiplexer–concentrator, all the terminals can be carried by a single trunk with a

I. System comparison

System	Number of Channels	Probability of a Message in Error*
Direct connection	15	1.000×10^{-5}
TDM	8	1.000×10^{-5}
Multiplexer–concentrator with 20-character buffer	1	1.001×10^{-5}

*Approximate error rates. The exact calculation of probability of error in a message would depend on the details of the transmission system.

FIGURE 6. Average delay vs. queue length for 15 input terminals and three utilization factors.



* It has recently come to the author's attention that a study similar to that in Ref. 29 has been made—using somewhat different assumptions—in a paper by W. W. Chu.[31]

negligible increase in error rate, if the buffer storage has a capacity of some 20 characters. (Characters, eight bits in size, are given four-bit addresses so that the channel utilization is 0.675.[*]) Table I provides a comparison of the systems.

This example is only an indication of what can be achieved by a very simple system under ideal circumstances. Efficiency, it must be added, is not by any means the only criterion of goodness for such a system. Ideally the entire system should be optimized, say from the point of view of cost, taking careful account of the traffic statistics. There is room for much effort in this area.

Conclusion

The purpose of this article is not so much to present a solution (it is not a new solution) as it is to encourage discourse and further work in the data-transmission area. Considerations indicate that there are tremendous gains that may sometimes be made in adding traffic considerations to the popular signal-design considerations, which represent the bulk of the current effort in data-transmission research.

When pulse-code modulation becomes widely available, thereby increasing the data-transmission capacity of the voice channel by a factor of roughly ten, the mismatch between low-speed terminals and channel capacity will become even greater and the use of multiplexing–concentrating techniques may well be even more desirable.

[*] The total number of message bits that each second flow into the trunk from the output of the 15 terminals is: 15 terminals × 1000 b/s per terminal × 0.06 utilization = 900 b/s. Since each eight-bit character is accompanied by an additional four-bit address, the trunk must carry an extra 450 b/s. The total trunk load is 1350 b/s whereas the total space available is 2000 b/s. The channel utilization is, therefore, 0.675.

The author is deeply indebted to many of his colleagues for exciting discussions on this subject. Particularly, the author wishes to thank H. R. Mueller, E. Port, and D. Seitzer.

REFERENCES

1. Becker, F. K., "An exploratory, multi-level vestigial side-band data terminal for use on high grade voice facilities," *Conf. Record First IEEE Annual Communications Conv.*, pp. 481–484, June 1965.

2. Critchlow, D. L., Dennard, R. H., and Hopner, E., "A vestigial-sideband, phase-reversal data transmission system," *IBM J.*, vol. 8, pp. 33–42, Jan. 1964.

3. Kretzmer, E., "Binary data communication by partial response transmission," *Conf. Record First IEEE Annual Communications Conv.*, pp. 451–455, June 1965.

4. Howson, R. D., "An analysis of the capabilities of polybinary data transmission," *IEEE Trans. Communication Technology*, vol. 13, pp. 312–319, Sept. 1965.

5. Schreiner, K. E., Funk, H. L., and Hopner, E., "Automatic distortion correction for efficient pulse transmission," *IBM J.*, vol. 9, pp. 20–30, Jan. 1965.

6. Rudin, H. R., "Automatic equalization using transversal filters," *IEEE Spectrum*, vol. 4, pp. 53–59, Jan. 1967.

7. Di Toro, M. J., "Communication in time-frequency spread media using adaptive equalization," *Proc. IEEE*, vol. 56, pp. 1653–1678, Oct. 1968.

8. Melas, C. M., "Reliable data transmission through noisy media—a systems approach," Conference Paper CP 61-377, AIEE Winter General Meeting, Feb. 1, 1961.

9. Burton, H. O., and Weldon, E. J., "An error control system for use with a high speed voiceband data set," *Conf. Record First IEEE Annual Communications Conv.*, pp. 489–490, June 1965.

10. Kohlenberg, A., "9600 bps—a magic speed for data transmission," *Telecommunications*, vol. 2, Nov. 1968.

11. Farrow, C. W., and Holzman, L. N., "Nationwide field trial performance of a multilevel vestigial-sideband data terminal for switched network voice channels," *Conf. Record 1968 IEEE Annual Communications Conv.*, pp. 782–787, June 1968.

12. Shannon, C. E., *The Mathematical Theory of Communication.* Urbana: University of Illinois Press, 1949.

13. Bennett, W. R., and Davey, J. R., *Data Transmission.* New York: McGraw-Hill, 1965.

14. Lucky, R. W., Salz, J., and Weldon, E. J., *Principles of Data Communication.* New York: McGraw-Hill, 1968.

15. Andrews, M. C., Cookson, B., Halsey, J. R., Lissandrello, G. J., Mueller, H. R., and Port, E., "A PCM-compatible switched data network study," *Conf. Record of Switching Techniques for Telecommunication Networks—London*, pp. 329–332, Apr. 1969.

16. Filipowsky, R. J., and Scherer, E. H., "Digital data transmission systems of the future," *IRE Trans. Communications Systems*, vol. CS-9, pp. 88–96, Mar. 1961.

17. Jackson, P. E., and Stubbs, C. D., "A study of multiaccess computer communications," *Proc. AFIPS 1969 Spring Joint Computer Conf.*, vol. 34, pp. 471–504, 1969.

18. Bacon, W. M., "Low speed data systems development in the U.S.A.," presented at 1969 Data Transmission Conference, Mannheim, Germany, Mar. 19–21.

19. Cornell, W. A., "The influence of data communications on switching systems," *Conf. Record of Switching Techniques for Telecommunication Networks—London*, pp. 342–345, Apr. 1969.

20. Hamsher, D. H., ed., *Communication System Engineering Handbook.* New York: McGraw-Hill, 1967.

21. Bell Telephone Laboratories, *Transmission Systems for Communications.* Winston-Salem, N.C.: Western Electric Company, Inc., 1964 (see especially Chap. 5).

22. Fultz, K. E., and Penick, D. B., "The T1 carrier system," *Bell System Tech. J.*, vol. 44, pp. 1405–1451, Sept. 1965.

23. Watson, J. J., III, "Concentrating and switching equipment for a real time multiple access communication system," *Conf. Record 1968 IEEE Internat'l Conf. on Communication*, pp. 123–128, June 1968.

24. Padegs, A., "The structure of system/360—part IV—Channel design considerations," *IBM Sys. J.*, vol. 3, no. 2, pp. 165–180, 1964.

25. Ossanna, J. F., Mikus, L. E., and Dunten, S. D., "Communications and input/output switching in a multiplex computing system," *Proc. Fall Joint Computer Conf.*, pp. 231–247, 1965.

26. Daley, E. A., and Scott, A. E., "IBM 7740 communication control system," *IEEE Conv. Record*, vol. 12, pt. 5, pp. 207–215, 1964.

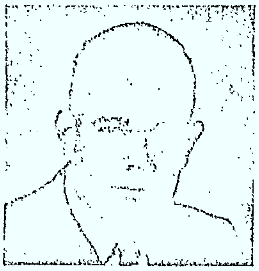
27. Drescher, J. E., and Zito, C. A., "The IBM 7741—a communications-oriented computer," *IEEE Conv. Record*, vol. 12, pt. 5, pp. 216–224, 1964.

28. Arnold, O. E., "Automatic polling by remote multiplexers," *Telecommunications*, vol. 3, pp. 17–19, June 1969.

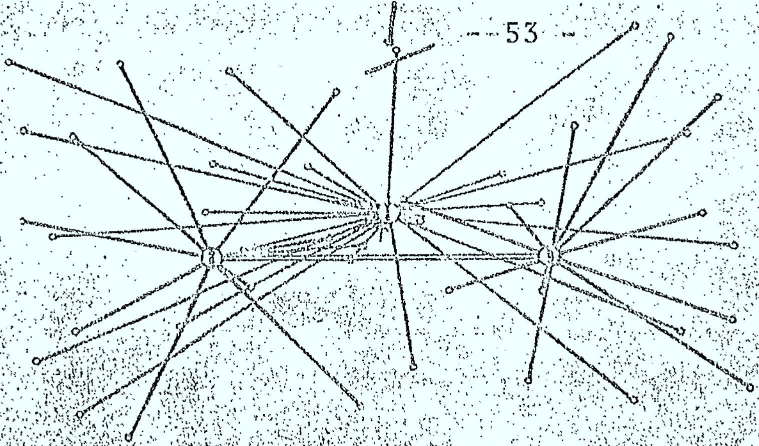
29. Rudin, H. R., "Statistical performance of a simple multiplexer-concentrator for communication channels," to be published.

30. Baskin, H. B., "The communications requirements of interactive computer graphics," presented at IEEE Internat'l Conf. on Communications, June 9–11, 1969.

31. Chu, W. W., "A study of the technique of asynchronous time division multiplexing for time-sharing computer communications," *Proc. Second Hawaii Internat'l Conf. on System Sciences*, pp. 607–610, Jan. 1969.



Harry Rudin (M) is currently working as a full-time consultant at IBM's European Research Laboratory in Zurich, Switzerland. His work is in the field of computer-related communications with emphasis on the application of traffic theory to the problem of data flow in communication networks. In his previous position at Bell Telephone Laboratories in Holmdel, N.J., he worked in the area of data communications, concentrating on automatic equalization techniques. From 1961 to 1964, Dr. Rudin served as an instructor in electrical engineering at Yale University. There, also, he received the bachelor of engineering, the master of engineering, and the doctor of engineering degrees in '58, '60, and '64, respectively. Dr. Rudin had served as a member of the Executive Committee of the IEEE Connecticut Section from 1962 to 1964.



Data communications

The telephone network that grew to meet the demand of voice communications is now the most convenient system for transmitting data—notwithstanding some technical drawbacks. Here are some basic facts about what is available and what must be done to put the existing facilities to use

Paul Hersch Associate Editor

When "digital computer" was an institutional rather than a household word, the machine required some very special personnel to cajole anything from its limited facilities. The operators came to the computer and spoon-fed it. And the computer's limited "brainpower" permitted only one operator at a time to tinker. Today, computers can be programmed so that almost anyone can be trained to access them—at least, in a limited way. Moreover, these electronic machines now have such massive central and peripheral memories that many people can use them simultaneously. Since it is impractical to have all of the computer users make a pilgrimage to the computer, the computer is made to come to the people that it serves. This is done via communication channels—predominantly in the telephone network. The various factors involved provide the substance for this primer article.

There is a certain irony associated with the development of telecommunications. Samuel Morse opened an era in the early 1840s. Then, some 30 years later, Alexander Bell developed a system that carried analog (voice) signals rather than cryptic dots and dashes; for the most part, it soon supplanted Morse's concept. What began as a digital communications network gave way to what was considered, until the past few years, a more convenient mode for communicating.

Now, at an ever-increasing rate, because of the growing presence and influence of the digital computer and a need to communicate with it from remote places on its own terms, there is a trend back to a Morse-like concept. However, most of the available channels for establishing the digital-communications links were tailored primarily for voice, and so it has been expedient to convert digital data into some form of analog equivalent in order to transmit the information. (Even telegraph communications invariably make use of voice-line communication channels.)

The situation is changing, and there is even some speculation that, since digital techniques are now viable, all information—including voice—inevitably will be transmitted in digital form. (It is not unusual even now for voice telephone conversations to travel at least a portion of their transmission path in digital form.) Although complete digital networks are now in the planning stage, they are unlikely to make much of an impact on data communications, let alone the entire communications realm, for at least the next five years. This discussion is about data communications now and in the near future and so it is, for the most part, the story of data transmission using presently available analog facilities.

This article, however, is concerned not with all facets of this very broad subject, but only with those systems that are fast becoming a significant factor in the scheme of modern business—communications involving data exchange with computers. Most of these systems employ common-carrier facilities—either ordinary switched-voice telephone channels or some private-line service leased from the carriers.

Because some readers will be more interested in particular areas of the subject, the article is divided into four main sections: basics, carrier systems, equipment, and system design. Each is relatively self-contained.

Basics

Symbol. in bo

Although digital communications emerged in its present form in the early 1950s, much of the terminology and basic knowledge derive from the old art of telegraphy. One result of the carryover is the variety of ways in which data-transmission rates are now given, and the confusion that has been created thereby.

For example, there are those who have trouble distinguishing between a baud and a bit. The baud—an old telegraph term—represents a basic rate of transmission in pulses per second. The amount of information that can be packed into each baud is represented by the number of bits per baud. With many data-communications systems, primarily the slower-speed systems, it has been customary to use binary* (two-state) signaling and, in this instance, the baud and the bit rate are equivalent. Although the baud rate is limited theoretically to twice (and, practically, to only) the width in hertz of the bandwidth being transmitted, bit rate can be increased severalfold by nonbinary signaling techniques.

Whereas some people bandy bits and others bandy bauds, still others describe their systems as transmitting so many characters per second, or per minute.

At least one reason for adopting characters per unit time when specifying data-transmission rate grew out of a method for transmitting the data: in some systems, all of the bits that comprise a character are transmitted simultaneously (see multiplexing). One problem here, however, is that systems using the simultaneous-bit method may use a different number of bits—ranging from 5 to 11—to describe characters and so direct comparison of various transmission speeds in characters per second may be misleading.

Data-communication rates on voice telephone channels also generally are specified according to the broad categories of "slow," "medium," and "fast." Although there are no universally accepted definitions of these ranges, few would argue the point that data transmission at 300 b/s is slow. But one equipment manufacturer might consider 1200 b/s high speed; another might consider 1200 b/s moderate and place the fast crossover at 2400 b/s. In general, however, it appears that, as the data field matures and as techniques improve, there is a tendency to shift the slow/fast dividing line upward.

Aside from the fact that they may be misunderstood, data rates possibly may be misinterpreted. For instance, some equipment must operate with built-in delays; bits may be added for error protection (and, as such, are not message-information bits); and retransmission may by required in the event of errors. These factors, and others, reduce actual data-transmission rates.

Transmitting the data signal

Signals from computers and most other data terminals are generally simple on-off waveforms (called baseband). Most terminals receiving the data message also operate with baseband signals. It would be convenient if the signals were transmitted in the form in which they are generated. This can be done, and is common practice if the receiver and transmitter terminal are linked by short sections of connecting cable.

But, for the most part, the communication links available today have not been designed to carry baseband signals. If the data are to be moved from one section of the country to another, the baseband signals must be translated, through modulation, into an ac form at the proper frequency for transmission. There are three options, as shown in Fig. 1:

- The baseband signal can be used to control the amplitude of a sine-wave carrier. This contrivance is amplitude modulation or, as sometimes termed in the data field, amplitude shift keying (ASK).
- The signal can be used to control the frequency of a generated signal—better known in the data-communications field as frequency shift keying (FSK).
- The signal can be used to regulate the phase of a generated frequency—more often called phase shift keying (PSK).

All of the modulation methods are used, although ASK and PSK are the most popular for high-speed transmission of data, whereas FM is the overriding choice for low-speed applications.

ASK. The nature of amplitude modulation is such that two sidebands, each containing equivalent information, are produced. An analysis of ASK reveals that more efficient use of the communication channel is possible if either one or the other of the sidebands—rather than both—is transmitted.

In general, a double-sideband system must have a bandwidth equal to at least twice the baud rate; the single-sideband system is about twice as efficient in its use of bandwidth.

To demodulate ASK, it is necessary for the receiver to "work against" some reference carrier. The reference is intrinsic to double-sideband systems; single sideband requires the precise reinsertion of such a reference. Moreover, single sideband is extremely sensitive to channel impairments. As a compromise measure it has become common practice to transmit all of one sideband and a vestigial part of the other. Although it is possible to extract carrier information at the receiver using pure vestigial sideband (VSB) techniques, for practical purposes, reference tones usually are added at the extreme ends of the available bandwidth.

It should be noted that VSB requires expensive techniques and care must be exercised to ensure careful system adjustment. But VSB is among the most useful techniques for implementing high-speed data transmission over band-limited channels because good SNR is possible with a minimum expenditure of bandwidth.

FSK. Frequency modulation is angle modulation in which the instantaneous frequency deviation is proportional to the modulating voltage.

The FSK technique operates by having two or more different frequencies transmitted within the bandwidth of the communications channel in order to convey two or more levels of information.

Since amplitude excursions are unimportant for detecting an FM wave, frequency modulation is relatively immune to disturbances that create errors in an amplitude-modulated system. However, FSK, like double-sideband modulation, limits a channel's bit-carrying rate

* Also called mark-space.

3

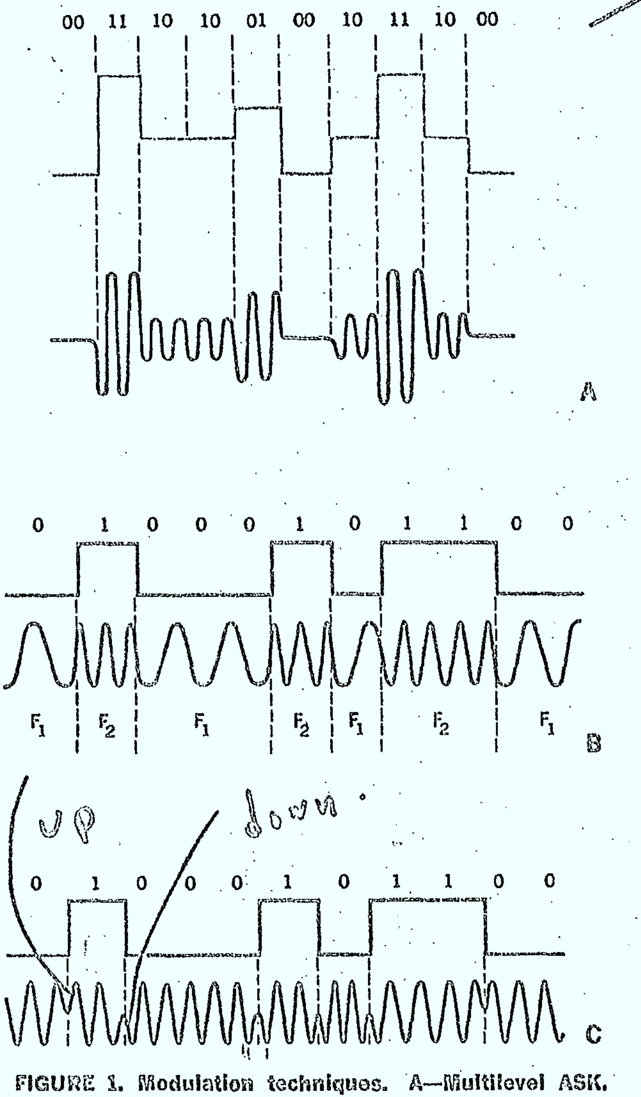


FIGURE 1. Modulation techniques. A—Multilevel ASK. B—FSK. C—PSK.

to about the frequency of the bandwidth. (Capacity varies somewhat with the degree, or index, of modulation.)

FSK might be termed a "quick and dirty" approach to transmitting data—so long as data rates are less than about 1200 b/s and bandwidth conservation is not a factor. It is "rugged" in the sense that it has good tolerance to many kinds of channel impairments and it offers the very attractive advantage of low-cost design.

PSK. In phase modulation, as the term implies, the phase of the transmitted carrier is altered to conform with the information being conveyed. In the simplest system, the phases used are 180 degrees apart; when more than two levels of information are carried, additional phase differences are added—usually in multiples of two (thus making the difference between existing phase shifts smaller). There can be a coherence problem and a reference must be obtained by the receiver. Alternatively, this problem can be overcome by using differential PSK (DPSK). Here, each successive phase change is made relative to the last existing phase rather than relative to some standard. Although a slight SNR penalty is incurred, DPSK is less susceptible to phase jitter and other channel disturbances. PSK, for highest-speed systems, can be complemented with amplitude modulation. When this procedure is followed, results are said to be comparable to those obtained with VSB.

M-ary techniques. Most low-speed data-communication systems operate on a binary coding principle. In such instances, the rate at which bits are transmitted is limited to about the available bandwidth. However, if the level of noise and other signal distortions permit, more than a dual-signal amplitude level, or more than opposite-signal phases of a PSK signal, can be transmitted. (See Fig. 1.) In this way, each baud carries with it more than one bit of information—more specifically, the logarithm to the base two of the M amplitude or phase options.[1] These M-ary techniques can also be used with FSK; however, M-ary ASK and PSK signaling is accomplished without increasing bandwidth, whereas M-ary FSK usually buys added bits at the expense of bandwidth. M-ary techniques using ASK-PSK or VSB have been used to build systems capable of data-transmission rates as high as 14 400 b/s over Bell System leased, conditioned voice-grade lines.

Although the type of modulated data signals that the communication channel carries is an important consideration, so are the ways in which the channel is used to interact the transmitter and the receiver equipment.

The modes of transmission

There are two transmission modes: synchronous and asynchronous (see Fig. 2). The former method feeds a continuous stream of data, which contains information immediately relevant to a message, along with some coding as to how the message is to be acquired and reconstructed. The latter method is start–stop in nature and either the data stream, the interval between message streams, or both, may occur irregularly.

Some of the advantages favoring one system over another are obvious; others are not. For example, the asynchronous system wastes bits because a start bit and a stop bit are added to each character, thus detracting more from the real message rate than with synchronous transmission in which simple framing patterns are applied to long blocks of characters. It follows that the synchronous system, lacking this constant reference feature, is the choice for highest transmission rates. On the other hand, asynchronous equipments can usually "talk" with one another over a wide range of data rates and, in this sense, compatible systems may be more easily organized from diverse pieces of equipment. The asynchronous systems also may prove more adaptable in "acclimating" to prevailing channel capacities. The dial-up phone network, for example, may provide a suitable link for operation at a particular data rate in one instance. On the next call-up, the connection may be suitable for only a portion of this rate.

Synchronization. When information is transmitted, there obviously must be some way for the receiver to "know" at what instant of time it should look for the message bits. Synchronization has two aspects: (1) acquisition—that is, the process of establishing sync—and (2) tracking, or the process of correcting for sync drift. There is no best way to achieve results. The chosen method will depend upon economics, total-system characteristics, and technological tradeoff.

- bit sync
- framing -

In all but the slowest systems, a series of pulses is generated at the transmitter to establish sync. The series continues until sync is affirmed—if there is an answer-back facility—or as long as is deemed necessary if no such facility exists.

There are several methods for maintaining sync. If the system is throughputting data at precisely defined, regular intervals, then a continuous clock message, transmitted in a portion of the radio spectrum different from that of the data stream, and filtered out at the receiver, can be transmitted. With such a system, it is unnecessary for the receiver to contain a clock.

Alternatively, sync can be maintained by extracting the necessary clock frequency from the data signal itself. The receiver then keeps its own clock locked to this derived frequency or, in the absence of a transmitted signal, the receiver clock is able to "freewheel" for a certain specified duration.

Although data systems operating above 100 baud usually require a continuous synchronizing system, continuous sync is dispensable for telegraph-speed start–stop operation. The receiver for such low-speed systems is resynchronized at the beginning of each character; then a simple clock in the receiver merely keys the sampling rate for the forthcoming bits of character.

Detection of the start signal in this telegraph-type system is highly influenced by noise. If the start pulse is missed, the receiver will substitute a subsequent character bit for it and entire character trains will be garbled until sync is reconfirmed.

Parallel or serial. In addition to a choice of synchronization, the data user may choose to transmit either in a serial or parallel mode. Selection usually depends on the original format of the data to be sent, although it also may depend on optimizing channel use.

For parallel transmission, the channel is broken into many subchannels of narrower bandwidth—that is, frequency-division multiplexed; see Fig. 3. Typically, each bit of a character is transmitted over a separate narrow channel. The bit rate of each small channel is reduced by a factor equal to the number of channel segments. In serial transmission, on the other hand, each bit of an individual character is transmitted sequentially over a single channel.

Channel direction. The user of data-transmission equipment can decide either to send *or* receive (simplex), or to send *and* receive (duplex). In the latter instance, he may elect to send and receive on an alternating basis (half duplex) or simultaneously (full duplex). Half-duplex transmission can be accomplished over two-wire service; full-duplex transmission can be achieved either with two wires (using a device called a hybrid junction) or four wires. Most computer terminals operate in the half-duplex mode.

Codes

Since data systems transmit binary information, it is necessary to structure an alphanumeric character set by arranging the zeros and ones so that different formations correspond to different numbers of letters. With the most straightforward combinatorial codes, there are 2^n possible alphanumeric representations in an n-bit word. For example, a five-bit character set can be used to express 2^5, or 32, characters or numbers.

Several systems for translating the alphabet and other symbols into a series of binary signals are now in use.

The Baudot code is a telegraph code. Each character is represented by five signaling bits. As just explained, five bits, ordinarily, can convey only 32 different characters; by using two of the 32 combinations as a case-shift signal, Baudot is expanded to 60-character capability.

The American Standard Code for Information Interchange (ASCII) is rapidly becoming a standard of the data-communications field. The system makes use of seven character-identifying bits, which often are supplemented with a parity bit for error-checking purposes.

The BCD, or binary coded decimal, code comprises four-bit blocks. Alphanumeric characters can be transmitted in two ways: (1) an extended BCD in which two so-called zone bits plus a parity bit are added to the basic four-bit block; or (2) two blocked-together four-bit blocks—one for zone and the other for numeric control. Most magnetic tape units make use of extended BDC; computers use a form of blocked-together BCD called extended binary-coded decimal interchange (EBCDIC).

There is also a Hollerith code that is used extensively with computer card readers. It too makes use of zone and digit areas. An alphabet character is coded as two holes in a 12-position column; a digit is represented by a single hole.

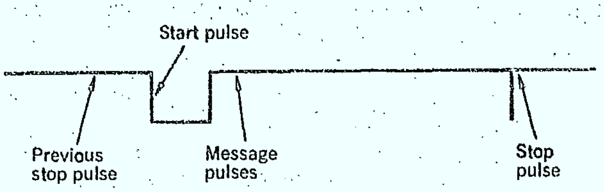
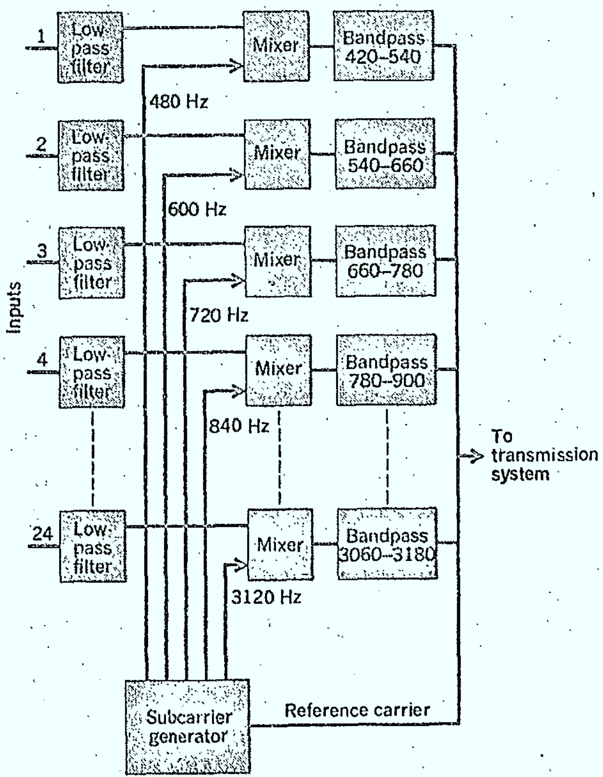


FIGURE 2. Simple asynchronous receiver keys on start pulse, then samples the remainder of the message at an internally preset rate.

FIGURE 3. Typical frequency-multiplexing scheme.



5

Errors

Error rate is a factor commanding considerable concern in the data field. The most basic sort of data receiver must decide only whether the signal represents a zero or a one. With the more sophisticated *M*-ary equipment, a decision also must be made among more than two alternatives.

Although many factors influence error creation, noise is the chief culprit. If noise adds to, or detracts from, the signal being transmitted—depending upon the longevity and severity of the noise, and the elegance of the hardware—bits, characters, or entire blocks of information can be destroyed; see Fig. 4.

Noise can appear in two forms: background and burst. The former, commonly called Gaussian noise, is always present since it is a creation of such phenomenons as the random (thermal) motion of electrons throughout the system. (Its effect is heard as a hiss during an ordinary telephone conversation.) The other (burst) noise results from such events as lightning strikes and dropouts or crosstalk from making and breaking of channel circuits.

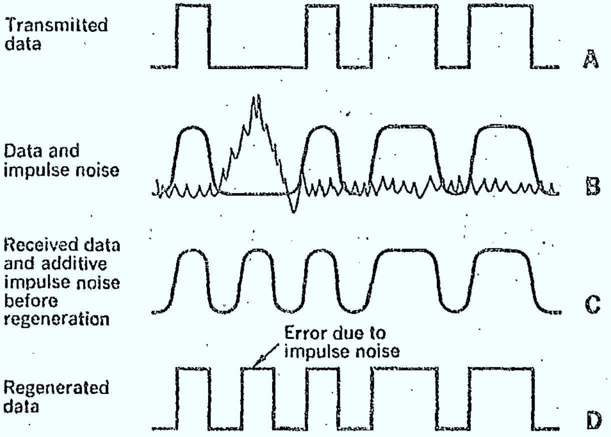
Burst-noise effects are typified by abnormally high signal transients or, alternatively, the loss or near-loss of received signal for a time span upwards of several milliseconds.

The errors caused by burst noise tend to cluster in groups, whereas the errors due to background noise most often occur in widely scattered, individual bits. Unless coding against burst noise is used, it tends to be the dominant cause of errors. When coding against bursts is employed, then both forms of noise have to be considered.

Although almost all errors in data transmission are caused by noise, the susceptibility of a system to these noise-induced errors often is determined by other signal distortions that occur during transmission. The most important of these signal distortions is caused by the imperfect frequency response (amplitude and phase distortion) of the typical telephone channel. This distortion causes a tendency for pulses to be distorted or to be smeared into one another during transmission. Other sources of signal distortion include frequency offset, nonlinearities, and phase jitter.

Most errors that occur with low-speed data-transmission systems are caused by burst noise. Other error-inducing factors attain significance as the data rate grows.

FIGURE 4. An impulse spike (B) interjects a "one" (C,D) where a "zero" had been (A).


Transmitted data
A
Data and impulse noise
B
Received data and additive impulse noise before regeneration
C
Error due to impulse noise
Regenerated data
D

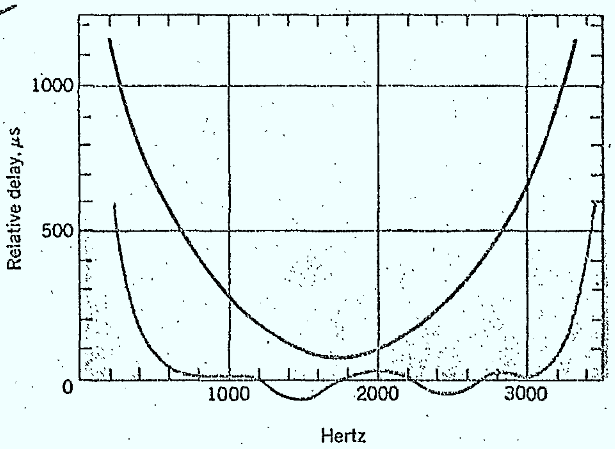


FIGURE 5. Typical loaded-line delay curves: (black) before equalization and (color) after compensation.

Thus, the highest-speed systems often are limited by phase jitter and nonlinear channel characteristics.

Improving the transmission system

The problem of line amplitude and phase distortion can be attacked in several ways. The communication line can be measured and elements inserted to compensate for differing distortions at different frequencies; see Fig. 5. When channel characteristics are variable, the receiver equipment can be supplied with variable circuit components to compensate for nonuniform channel attenuation and time delay. This procedure can be accomplished manually or by the equipment as an adaptive process. There also are compromise equalizers that can be added to the front end of a receiver and/or transmitter and thereby provide nominal improvement of a very bad channel; however, their use sometimes will degrade, rather than improve, an existing good channel.

To overcome Gaussian noise, in theory, a message only need be transmitted with sufficient power. For example, in a two-level (on–off) system the SNR at the receiver must be at least 10 dB in order to reduce transmission errors to a generally acceptable level. ("Acceptable" has different meanings for different systems, but many of the systems operating over common-carrier facilities tolerate one error in every 10^4 or 10^5 bits.)

Raising the transmitted power level is not the solution to the noise problem as it would be in a radio system. For telephone-line data transmission, the power is limited by crosstalk considerations that come about because of the potential overloading of wide-band amplifiers used for carrier transmission. Also, the patterns of burst noise would be more or less unaffected by system power level.

Thus, in any real system, some errors are inevitable. These errors can be ignored if they subsequently can be deciphered. In other instances, it is important to provide some arrangement that at least detects received errors, so that a retransmission can be requested. In still other instances, it is not enough simply to detect; automatic correction is required.

Both error recognition and error correction are possible if the original data message is supplemented with an error-coding scheme. Obviously, error correction requires more complex codes than simple error recognition.

Error control

It is worthwhile to note here some of the techniques that are used in an effort to note and/or control the number of errors.

To control errors, some sort of information about the "appearance" of the original message block is incorporated into the original message.[2]

Parity codes. The simplest procedures are exemplified by parity codes such as ASCII that contain an added bit, which is a function of the number of ones or zeros in the character-bit string; that is, parity depends on whether the number is odd or even. This type of protection can break down if errors occur in pairs. Therefore, the technique can be upgraded by storing each character in a buffer "one under the other," whereupon parity is checked both horizontally and vertically. It is also possible to add another dimension to this sort of parity check: the bits of the stored character matrix can be parity-checked on a diagonal.

· long parity

There are other, more powerful codes that use more than single-parity checks. Their facility has been made possible by the development of low-cost shift registers and modulo-2 adders. One of the most extensively used of these codes is the cyclic type.

The code is enabled by adding a number of zeros (which depends on the anticipated noise) to the original code character and then dividing the enhanced character by a suitably chosen series of ones and zeros. The number of bits in the divisor is equal to the number of zeros appended to the original coded character.

Then, the enhanced character and the remainder after division are transmitted. At the receiver, the enhanced code is divided by the same series of bits that comprise the divisor at the transmitting end. The remainder derived at the receiving station is compared with that received from the transmitter.

Constant-ratio codes are yet another variety for determining when an error has been committed during the passage of a message. Each character is composed of a constant number of ones and zeros. As long as a switch of a one to a zero and a zero to a one doesn't occur simultaneously, the code will detect the error.

ARQ (ACK/NAK). Using these coding schemes, the data communicator increases his chances of detecting errors—often to the extent of picking out all errors exceeding one in 10^9. However, once alerted to the error, the data user must have some way of discarding the errors and substituting correct information. A method for doing this is called ARQ (automatic repeat request) confirming a good message or relating that an error has occurred. On full-duplex lines, this answer back can be performed while new information continues to be received. On the more common half-duplex lines, the *line control discipline*[3] causes the direction of transmission to be reversed at the end of each "block" of many characters, so as to send to the transmitter either an ACK (positive acknowledgement) if no error is detected, or a NAK (negative acknowledgment—that is, repeat request) if the block is in error.

In contrast to ACK/NAK is forward error correction (FEC)—that is, the coded message, if it is received in error, is corrected at the receiving station.

The codes previously mentioned for simple error detection also can be used for error correction—if they provide suitable redundancy.

Other codes used for forward error correction are of the convolutional class. These do not have a block structure and, in some cases, are easier to decode than the codes hitherto discussed. In principle, such codes alternate parity bits (computed on the basis of the content of a previous segment of the message) with one or more message bits. Parity bits also are computed at the receiver station and compared with the transmitted parity bits (in much the same way that was described for the cyclic code). This process is performed continuously.

Two basic methods generally are used for decoding convolutional codes—threshold and sequential. The former is easier to implement; the latter is more powerful but requires relatively complex circuitry. Whereas threshold decoding manipulates the message and parity bits and computes an adjustment within a particular length of message, sequential decoding attempts to best fit the received data sequence with a tentative correct one postulated on the basis of a past history of received data. When there is no immediate match, a search begins—backward and forward—along a prescribed coding tree.

Carrier systems

It is axiomatic that the key element of any data-transmission system is its transmission channel.

The data user has options, and several suboptions, on the systems at his disposal. Basically, he can build a network of his own; or he can subscribe to a service supplied by some communications carrier.

A private long-haul carrier system may be constructed for various reasons. Often such a system is a logical extension of existing, limited facilities (as the signaling facilities used by the railroads). Sometimes it must be constructed because no other (or only a limited common-carrier) service exists. Many fine HF and microwave systems are on the market, and recent technological advances have opened a broad, previously little used, low-frequency spectrum. However, the data user who chooses to be independent of the common carriers must deal with problems concerning rights of way, interference with other carriers, and careful evaluation of the best system over a prolonged time span. (For example, cable may be more costly initially but prove to be a cost saver over microwave or RF equipment in the long run.)

Usually, the reasons for employing private facilities are insufficient for the majority of data users; and so the remainder of this section is devoted to common-carrier facilities.

The Bell System

At the moment, the prospective user of communication channels can obtain network* facilities from three sources: Microwave Communications, Incorporated; the

*Companies other than MCI have filed with the FCC. However, MCI anticipates formal FCC approval to occur at about the time of publication of this article.

Bell Telephone System; and Western Union (which, at the moment, itself is predominantly dependent on leasing Bell channels).

By far the largest network is that of the Bell System—a vast 300-million-channel-mile system of loaded cable, open wire, coaxial cable, and microwave facilities, with most of the long-haul transmissions handled by microwave and coax. (Future plans call for the addition of waveguide and satellite facilities.) The largest share of the system is designed for analog transmission and the major portion of it is for voice communication. Several broadband facilities are available and straight digital offerings (i.e., no D/A conversions), though quite limited at the moment, are scheduled to become commonplace within the next two or three years.

Because of economic factors, the Bell network has been built so that each improvement, in general, has been compatible with existing facilities. The network includes an assortment of switches (for example, step switches, crossbar switches, and computer-driven crosspoint switches, such as the Bell ESS No. 1) and other devices. A point-to-point connection can comprise a large number of diverse elements.

Dialed and leased lines. Bell offers two options—leased line and direct distance dialing. With DDD, no special line preparation is made (other than from the customer to the local switching office comprising an overall connection) and the customer must take whatever circuit is made available. Sometimes the connection will be relatively direct and good; other times it will be quite circuitous—generally with degraded results. The data user may establish a connection within a matter of a few seconds. Chances are, however, that the average connection will consume 10–20 seconds; during peak periods of traffic, it may take even longer.

In general, because the quality of the line cannot be optimized on DDD, the data user is restricted to message rates not greater than 4800 b/s.* Whether the user will want this maximum rate is another matter; he must consider tradeoffs in the cost of equipment to achieve suitable transmission at that speed, and other factors.

If the communicator decides that he would rather have a more reliable channel—one that is always available and one in which he can have relative confidence with regard to performance, or if he can't wait to gain line access—he can lease a private line. Moreover, he can ask the telephone company to condition the line according to his sending-rate needs. Conditioning is available on three levels, C-1 (least), 2, and 4. These correct for the amplitude and phase characteristic of the line.

It is important to note with respect to line conditioning that these corrections may apply only for a single point-to-point hookup. In a multipoint situation, for example, where there is an intermediate station between end stations, it is not possible to obtain C-4 conditioning at the present time. The result of C-4 conditioning in such an environment is C-2 conditioning. In fact, any conditioning will be degraded a notch when applied to a multipoint facility.

The user of the leased line might assume that the same line is constantly at his disposal. This need not be true. Changes sometimes must be made in routing because of facility outages. Moreover, switching arrangements may be altered at the local switching office. Because each level of C conditioning is performed within certain limits, if there is a line switchover the user may be aware of an overall conditioning change within the conditioning range due to altered circuit parameters.

Any line fluctuations still present must remain unresolved or special automatic compensating electronics must be incorporated with the equipment used, either by the customer or manufacturer.

As previously mentioned, the majority of DDD and leased lines are voice- (or lower-) grade analog types. Available services using these DDD facilities include low-speed (to 150-b/s) TWX, wide-area telephone service† (WATS) on either a part-time (ten hours minimum) or a full-time basis, and dial-up voice-grade single-time-rate facilities. The customer has the option of using Bell-supplied equipment and/or service, or can use his own equipment through a Bell line-protection device (data access arrangement).

Private-line offerings of facilities that are voiceband wide, or narrower, include narrow-band telemetry and teletypewriter services (to 150 b/s), voice band, and a Data Line Concentrator System.

Up to the present time, the Bell System does not require the data-access arrangement on its private lines that it does for the DDD network. The customer can hook directly into the Bell system. This situation is expected to change, however, within the next few months.

In addition to the offerings just described, the Bell System provides wide-band facilities—both of a private-line and dial-up nature. Of the private-line offerings, there are Telpak C and D and Series 11 000. The last two provide bandwidths up to 240 kHz (although part of this bandwidth is reserved for telephone company use). Series 11 000 service is of a limited nature (seven states, 47 terminal points) but those customers who have access to it have more benefits than Telpak users. (For example, they can bring in shared users without restrictions.) Rates are also lower for 11 000 than for other comparable Bell wide-band offerings.

Another wide-band offering on the dial-up network is a still-experimental Data-Phone 50, which provides 50-kb/s data rates between certain cities.

Digital carrier facilities. At the outset of this article mention was made of transmitting digital information over communication channels without digital-to-analog conversion. Bell, at the present time, does provide digital data offerings, although they are limited and have not been tariffed. The company makes this series available over its T1 system, which is a digital carrier currently used for voice transmission via pulse code modulation on repeatered cable pairs. (In pulse-code modulation, the voice signal is sampled at a certain rate—Bell and others use a frequency of 8000 samples per second—and the signal level is then coded into digital pulses.) The first such system, T1/D1, went into operation in 1962. Each cable carries 24 voice channels, each of which requires 64 kb/s in digital form (8000 × 8 bits). The T1 carrier is not suitable for toll-call network use since the range of the system is limited to less than 80 km. This service was

*Equipment is available that is said to operate up to 5400 b/s on DDD

†The customer may call out and/or receive incoming calls on a flat-rate basis between his station and any other station within one of five "radii" of his choosing.

originally developed to relieve the shortage in urban central-exchange-to-central-exchange usage of copper-wire pairs, each of which carries only one voice channel in normal analog fashion but with digital 1.54-megabaud PCM can carry 24 voice channels (24 × 64 kb = 1.54 Mb). The system has been deployed about the U.S. in isolated pockets and a total of 500 000 channels is available. Although not originally developed for data communication, it is clear that this service is an important resource to be tapped for future data traffic.

In the next few years Bell expects to put an improved version of the D/T system into operation. This T2 system will process and transmit the equivalent of 96 voice-grade channels and, additionally, will be suitable for toll calls. The area that the improved service is to cover also will be extended to a range of up to about 800 km.

These digital carrier facilities, together with other new types of long-haul carriers, will be able to form the backbone of an all-digital data network in the near future. American Telephone and Telegraph Company has announced plans for a private-line data network, which it calls a data pipe, to be available late in 1973. Customers using this projected data network will be time-multiplexed onto the digital-carrier facilities. The economy of operation of such a system should be obvious since each voice line occupies 56 kb of T1 carrier capacity. (Present analog voice facilities are limited to 14.4 kb.)

Another offering that is expected to grow is data service that will become available as an adjunct of Picturephone®. The Picturephone channels will permit information transfer at speeds up to 1.3 Mb/s.

Datran

In many respects, a system proposed by Datran (Data Transmission Company) in November 1969, which is pending approval before the FCC, is similar to a digital network that Bell has been planning.

If and when Datran is approved as a common carrier,[4] it, in the sense that it will be an alternative to Bell service, will be in a league with the recently inaugurated Microwave Communications of America MCI offerings.

Datran's proposed offering represents a direct-distance-dial service (designed) for transmitting data without intermediary analog conversion.

The Datran communication net is based on 4800-b/s-capacity channels of less than 2-kHz effective bandwidth. These channels may be tied together for transmitting 9600 or 14 400 b/s; or the basic channel can be parceled into 150-b/s subchannels. A 48 000-b/s leased-only service also has been proposed. These four transmissions speeds constitute the options available to the Datran customer.

On the basis of surveys, Datran has adopted a specified number of channels for each of the 35 cities it initially proposes to service. (The system readily could be expanded to encompass 53 cities, according to Datran.) As shown in Fig. 6, the Datran system will be all-microwave, except for some very short spurs.

The activity on the Datran network will be controlled by several regional offices, each of which will have control over as many as ten district offices. Connections for 1000 to 6000 terminals are said to be possible through the switching centers in each of the district offices. Customers within 80 km of district offices will be serviced directly via microwave; beyond this radius, line concentrators will be used. A special feature that Datran will provide is the feasibility of broadcasting any one message to up to six subscribers.

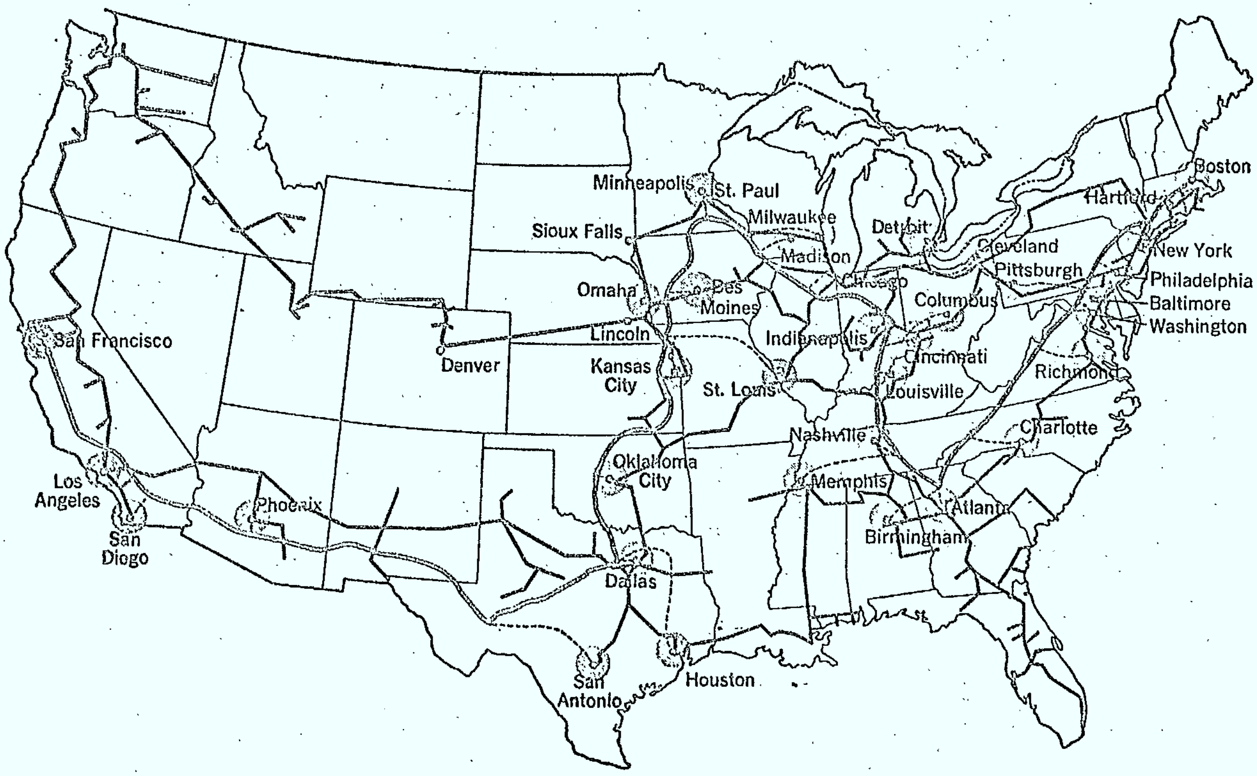


FIGURE 6. The proposed MCI carrier network (color) and the Datran data communication system (black).

Datran's lines presumably will enable data transmission with an error rate of only one in ten million. Datran also will provide and maintain customer equipment at the customer's option as part of its total service—in the manner of today's operating telephone companies. Planned features of the future include store-and-forward capability (see "System Design").

Datran emphasizes that connect time will be consummated in less than 3 seconds 99 percent of the time, assuming that the addressee is not engaged with another call. Moreover, it is said that a maximum of but three switching centers (at the outset, but only four regardless of expansion) will ever be involved in putting through a call.

An important economic aspect of data-communication carrier facilities concerns the shortest duration of a call for which the customer will be billed. The requirements of computer-to-computer, and particularly terminal-to-computer, communication are quite unlike those of voice; many inquiry–response conversations take only a few seconds whereas users of time-shared terminals or research computing may connect for tens of minutes at a time. The Bell System is considering a new 1-minute minimum billable time and Datran plans to use a 6-second figure.

Datran in its proposal to the FCC says that it expects to be ready for about 48 000 clients by 1975, assuming that FCC approval is forthcoming within the next several months.

MCI

MCI was conceived for reasons different from those offered by Datran.[5] It was said to be created as a private-line alternative to AT&T's offerings. The broadband transmission channels that it will use will be chopped into a multitude of smaller channels providing tailored analog and all-digital service.

The MCI network, as shown in Fig. 6, is made up of 17 separately owned, autonomous affiliates, with the parent company coordinating services and providing consultation. Although a private-line network, its operation is premised on the voice/data use ratio existing for Bell's DDD network—that is, 70 percent of the traffic is expected to be voice. Although the company anticipates the beginning of operations in 1971, the entire network would not be in operation before 1975.

As planned—and needs are constantly being reassessed—the final form of the MCI network will provide the equivalent of 100 000 separate voice-grade telephone channels in 1975. According to the original proposal, the system user was to have been responsible for getting his information to and from the central transmitting/receiving station.

It was MCI's contention that terminal-to-transmitter hookup could be accomplished by patching into Bell's network facilities. AT&T objected (unavailingly) that such a setup would only aggravate a situation that MCI said it was trying to eliminate—overcrowded central offices in (certain) urban areas. (Such a connection, it should be noted, seriously impairs one of MCI's most important reasons for being, namely, low-error-rate transmission.)

In light of the questionable practice of hooking MCI transmissions to existing telephone equipment, MCI has revised its original concept and a customer accordingly will have one of three options:

1. Hook to telephone facilities if convenient and practicable.
2. Provide his own link between his sites and MCI's transmitter/receiver.
3. Ask MCI to install the short-range hookup between the central microwave tower and the customers' sites.

In the latter two instances, the customer could choose among cable, wire-pair, microwave, and optical links, as well as other methods, depending on location needs and strictures.

A novel feature of the MCI concept is that it provides the user with only as much long-haul bandwidth as he needs for one- and/or two-way communication. In all, 138 differently sized channels, offering from 200 Hz to 0.96 MHz, are to be available for either full- or part-time data-communications use. Radio-frequency channels will be offered in seven basic groups.

Western Union

Western Union's offerings include Telex and, starting in March, TWX. Much of the network that the utility now uses is, itself, leased Bell facilities. However, in the past several months, WU has made two proposals that will add considerably to its "in-house" capability. One of these additions, a 700-km hybrid extension to its existing "transcontinental" facilities, represents WU's first extension into the South—serving a region between Atlanta and Cincinnati. This extension will provide for 165 low-speed digital channels and 216 voice channels. The other system, forming a network between the cities of New York and Chicago and including Philadelphia, Cincinnati, Cleveland, and Detroit, among others, will be an all-digital system with a 20-Mb/s capacity.

The equipment

A computer and its peripheral devices connect into the communication system by means of a transmission control unit or "front-end processor," as shown in Fig. 7. In addition, there are at least two components at either end of a communications channel: (1) a terminal for generating outgoing information and/or recording incoming information; (2) a device for conditioning the signal for transmission over the communication facilities and/or conditioning an incoming message for acceptance by the terminal. In addition, there will be some sort of auxiliary equipment contained in the terminal or situated between the terminal and conditioning equipment for error detecting, delaying, and/or grouping the data, and for handling line turnaround and other demands of the line control discipline.

Modems

The equipment that conditions the incoming or outgoing signals is known by a variety of terms. It has been called a line adapter, data set, and modulator. One of

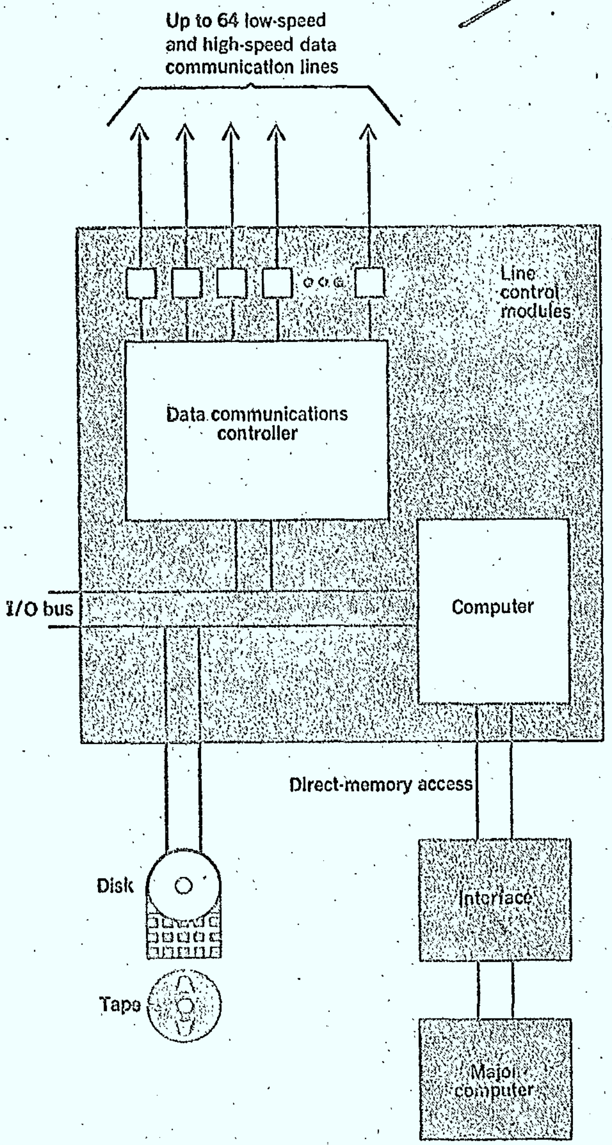

Up to 64 low-speed and high-speed data communication lines
Line control modules
Data communications controller
I/O bus
Computer
Direct-memory access
Disk
Tape
Interface
Major computer

FIGURE 7. The computer must operate through a transmission control unit when it is a part of a data-communication system.

the more popular terms is "modem" (signifying its modulation/demodulation functions). A modem by any name has one primary task— to take a dc signal and convert it into an ac representation using FSK, ASK, PSK, or combinations thereof. The modem also may have such other features as automatic equalization, provision for full-duplex operation, or operation with a second channel having a fraction of the bandwidth of the main channel. The modem optionally may have a built-in encoder/decoder.

Most of the modems for use on DDD voice-grade facilities are available with certain standard data-rate capabilities.[6] These are 75, 150, 300, 600, 1200, 1800, 2400, 3600, and 5400 b/s. (Some companies offer intermediate speeds.) Some modems can be operated at speeds higher than rated; others can't. Company standard practices vary.

If the modem will be used with a leased line, it often will be capable of operating at better than nominal DDD voice-grade ratings. And if modems are specially purchased for private-channel use, they can be obtained with operating speeds to 14 400 b/s in very special cases.

Many low-speed asynchronous data sets are compatible with one another regardless of their source of manufacture (depending, of course, on whether they use the same modulation techniques and operating frequencies, among other factors).

On the other hand, high-speed modems invariably are intended for synchronous operation. These sets are designed for specific operating speeds, and when a set has more than one operating speed, they all are usually multiples of some basic clock frequency. Most high-speed units today will lock receiver and transmitter together in a matter of seconds. Some equipment will achieve acquisition in a matter of tens of milliseconds. There is steady pressure to improve the figures, since for frequent line turnarounds or frequent short calls, the modem "settling time" consumes as much time as the message.

Apparently, there's a rough rule-of-thumb formula that the cost of a modem is about a dollar per baud. This estimate accounts for the extras that are usually added as rated speed is increased to give the device its maximum utility.

Some modems will accept several channels, but in this respect they function as several independent modems with outputs over several lines and, therefore, are not concentrators (which we shall discuss shortly). Other features, which may or may not be to the user's advantage—depending on the complexity of his system, include provision for built-in testing and a voice/data option.

A modem either may be hard-wired to the communications channel or tied to the channel acoustically. The acoustic-coupled modem offers a degree of portability in that it can be used with any available telephone.[7] Acoustic modems readily can be used for telegraphic data rates. Most are rated at 300 b/s, but some manufacturers contend that their units will perform faster, and one manufacturer claims a rate of 1800 b/s for one of his devices. However, it generally is conceded that acoustic-coupled modems are prone to ambient acoustic noise and, because of the nonlinear and unpredictable nature of the acoustic-coupling interface, they are speed-limited.

A direct hookup to the communications channel is, therefore, preferable: it is a more secure and error-free arrangement. However, if the modem is not provided by the telephone company (and there is a trend toward terminal equipment that includes the modems internally) it may be necessary to interpose a so-called data-access unit between the modem and channel. This unit, which functions to protect the telephone network by limiting output signal level, is available as automatic or manual. The automatic version provides for automatic call and answer back.

For start–stop operation, performance is judged by the distortion in the transition of the output waveform. Terminals operating with nonsynchronous modems usually can operate with up to 20 percent distortion on simplex or half-duplex facilities. The distortion criterion takes on increased significance when the modem is operated full duplex. In this instance, incoming and outgoing signals can interfere and the tolerance of an asynchronous modem to distortion is reduced to about 7 percent. On the other hand, the performance of a synchronous modem is judged on its error rate with respect to SNR.

Terminals

The modem, as explained, serves as an interface between the line and the terminal. Among the terminal facilities are push-button Touch-Tone-type devices, printers, paper tape, magnetic tape, CRTs, and facsimile (FAX) machines.

The manner in which a terminal is applied usually falls into one of three classes:

With transaction terminals, the user is operating in a conversational mode with a computer. Messages in both directions tend to be quite brief.

Batch terminals output a string of transactions—usually on a scheduled or polled basis.

Retrieval terminals are used to gain access to computer files. The reply messages tend to be much longer than the inquiry messages.

Touch-Tone. One of the simplest and least costly terminals now in existence is the push-button telephone facility (in which no modem is required). This can be used with or without a card reader either as an integral part of the telephone or as an attachment, in which case it may be obtained from a source other than the telephone company and in versions that provide more than the standard 12 buttons. (These latter units require a data-access-arrangement interface; the phone company units do not.) Such a terminal in combination with a computer that provides recorded-voice answer back can perform simple query service. For example, the push-button-tone terminal can be and is used to carry out credit checks on credit-card holders.

Printers. Except for the Touch-Tone terminal, the printer is perhaps the most pervasive piece of terminal equipment in service. It can be obtained in "stripped" slow-speed teletypewriter (50 words/minute) versions, up through variants that spew 1200 or more words per minute onto a page.[8] Some are versatile enough to print out line diagrams and equations.

A fundamental difference between printers is their character set. Any given device may have only a subset of the full character set available with the code that it operates. Thus, some printers will provide upper and lower case; other printers using the same code may provide upper case only.

Most printers operate with an eight-bit ASCII code. Telex and some TWX equipment also use five-bit Baudot. The slower machines print character by character in the fashion of a typewriter. Faster machines will generate several characters, and even lines, at a time.

Recently, several machines that jet-spray two or more characters at a time onto the page have appeared on the market. There are also printers that use electrodes to "burn" an image onto a page and others that use thermal-transfer techniques.

The mechanical machines, although perhaps not as fast as the jet or electrode types, do have the advantage of being able to provide the user with multiple copies without running the added expense of a copying machine. The jet machine has an advantage over the electrode and thermal units in that no special paper is required.

A printer must be provided with some sort of control unit—either attached or built in—to code, power, and switch. This unit may serve one or a cluster of printer terminals.

The printer may be a receive-only model, or it may have a keyboard for send capability as well.

The printer's utility is increased significantly by adding auxiliary records. These devices store information on magnetic tape, paper tape, or punched cards and allow a user to (1) gather information during the day—inputting at typewriter speeds and editing as necessary—and then at rapid speeds transmitting at night; (2) receive at a rate too rapid for the printer—but economically more satisfactory for the user—for later "readout" at the printer's capacity.

Paper tape. Paper tape continues to be a communication staple. Although paper-tape units are slow (top recording rate is ten characters per second, whereas playback rate is a maximum of 100 characters per second), they do offer visual editing capability and are less costly than magnetic-tape machines.

Three paper-tape machine types are available. All punch holes in the tape, but some differ in readout procedure. Mechanical readers, which are reliable and low in cost, detect perforations by pins that pass through the tape. However, they are the slowest of the lot and induce excessive tape wear.

Pneumatic readers push air streams through the holes in the tape onto sensitive thermocouples. These units are faster and cause less wear than the mechanical devices, but do not provide the speeds possible with photoelectric machines, the third type of machine. Most photoelectric readers, moreover, employ friction rather than sprocket drive.

Tape
Cassettes

Magnetic tape. On the other hand, magnetic tape can record, typically, 100 times faster than paper tape. (Tape units operate at 36, 75, or 112.5 in/s at densities ranging from 100 to 3000 bits per inch, with typical densities being 200, 556, 800, or 1600 b/in.) It follows that magnetic tape also affords faster replay. Other magnetic-tape advantages include reusability and possibly more convenience in handling and storing.

CRTs. Although teletypewriters can be (and are) used for man–computer information exchange, they lack some of the versatility afforded by cathode-ray tubes. The CRT, for example, can be used with a "light pen" to make drawing changes. Perhaps the main use of the CRT, however, is in situations requiring a succession of rapid presentations—as, for example, in conjunction with airline reservations. Basic applications are sketched in Fig. 8.

Just as different printers operate using different principles, CRT operation covers a wide range. One basic group functions with a predetermined character set, code, and speed convention, and commonly is used with a keyboard input. Other CRTs "paint" each character and other information in a television-like manner. Some units combine character generation with graphical-display capability.

There are also differences in the way that CRTs refresh their display. Some are effectively storage tubes, others use some sort of memory (buffer) constantly to update the presentation. Using a buffer of this sort, it is possible to design displays that operate at telegraph rates (and, as a matter of fact, a CRT operating at 150 baud has recently been marketed).

Because it is annoying to the human user to have the screen fill up with characters at a rate slower than the natural reading rate, it is often necessary to present a complete display within a matter of some seconds. For this reason, CRTs usually operate at transmission rates

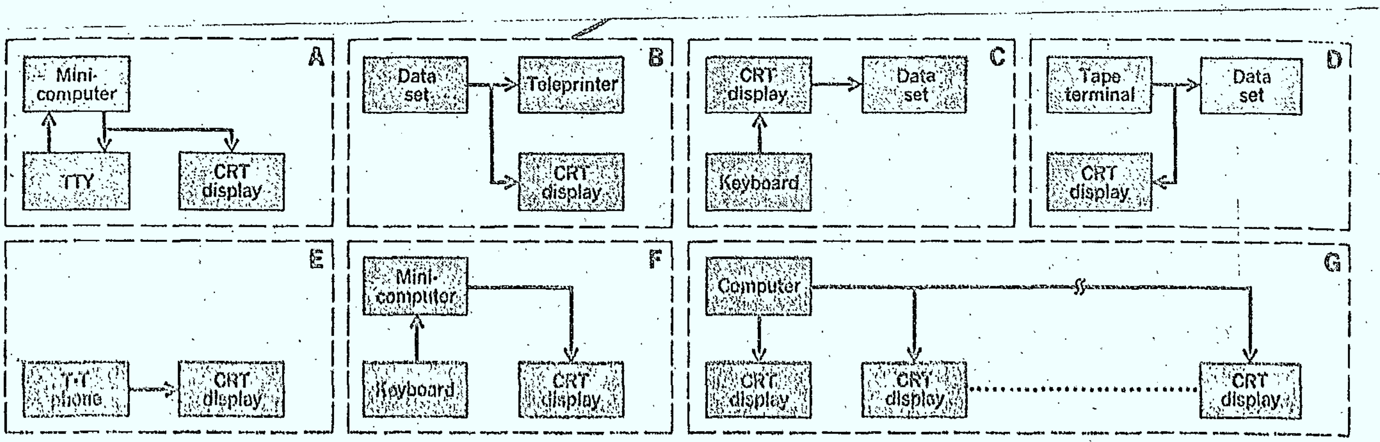


FIGURE 8. Typical CRT applications. A—Computer-output display. B—Time-share display. C—Keyboard entry and display. D-Communications monitor. E—Video answer back. F—Intelligent terminal. G—Party-line communication.

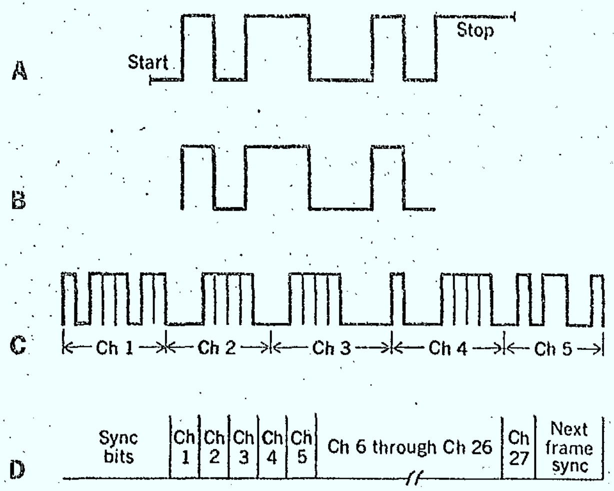


FIGURE 9. In a typical character-interleaved TDM system, the original message (A) first is stripped of its start and stop bits (B). Next, the character (channel 1) is put into its frame (C). Finally, the synchronizing information (D) is added.

exceeding 3600 b/s. This figure, as previously mentioned, just about taxes the capability of the switched Bell network—althought it is well within the capability of prepared leased lines.

For optimum utilization, the CRT terminal should be used together with a display control unit (DCU) that can preprocess and buffer the data, and thereby relieve the central processor of tedious and time-consuming "clerical" functions. (Some units can't be operated without a DCU; others can. Some DCUs serve a local cluster of units.) Whereas CRTs are useful for rapid, changing displays of information, most have the disadvantage of not providing for hard copy. However, at least one unit has recently been marketed that provides this capability. For those that don't, it is always possible to use both a CRT and a printer.

Last but not least, it is important to note the potential of Picturephone as a CRT device for displaying information from computers. At the present time, Picturephone is limited to intracity operation— in Pittsburgh and Chicago—over analog facilities. However, plans are that, by 1972, a mod 2 system will be instituted that will operate over digital facilities on an intercity basis. Customers of the service will therefore have a particularly suitable combination of equipment and transmission facilities for accessing computers at a very high data rate.

Another piece of equipment useful for data communication is the facsimile (FAX) machine. This sort of equipment has been in use for a considerable time, but, until recently, was the beneficiary of little technological improvement.

Most machines still transmit a standard 8- by 11-inch page within six minutes with a definition of some 100 lines per inch over voice-grade switched-network channels. In the past few years, some machines have been made available that provide page transmissions in about half the time using ordinary telephone facilities, either by sacrificing resolution or by improved information-packing techniques. For leased-line facilities it may be possible to transmit a "standard" page in about a minute.

Data concentrators and multiplexors

For reasons of economy, it is often desirable at some point in the system to combine many data channels into one, using either a multiplexor or a concentrator. A multiplexor is a device that simply adds the incoming channels together in some way—for example, by putting them side by side in frequency (frequency-division multiplexing, or FDM, depicted in Fig. 3) or in time slots (time-division multiplexing, or TDM, depicted in Fig. 9). In a multiplexor, the bandwidth of the single output channel is the sum of the input bandwidths. A concentrator, on the other hand, attempts to take advantage of the intermittent nature of the traffic in each incoming line to provide an output whose data flow is more constant with time, and therefore of smaller bandwidth than the sum of the input channel bandwidths.

Up to the present time, FDM remains the most widely used multiplexing scheme. FDM is easier to implement and, up to a certain number of terminals (the figure depends on many factors), is less costly than TDM. A typical FDM multiplexor might integrate a dozen 110-baud channels, or alternatively two 600-baud channels over one voice-grade line. Part of the usable FDM bandwidth must be relinquished to channel-guarding duty.

13

TDM, although it must give up message space for sync, framing, and channel-identifying bits, husbands the available channel to an extent greater than FDM.

And, most specifically, TDM represents the only practical multiplexing method over a true digital network since the packed-together bits needn't be put through a digital-to-analog converter.

There is a wide variety of data concentrators in use. These are often specially programmed data computers (commonly minicomputers) that may use for the buffering of incoming bit streams the main core memory plus possibly auxiliary storage devices such as disks.

System design

At a recent seminar sponsored by the National Electronics Conference, one lecturer stated that most of the data-transmission systems he had seen in his role as consultant could have been implemented for half their cost. Many knowledgeable people in the industry advise the prospective data user not to buy more than he can use. Often they will recommend that equipment be leased rather than purchased; and they will often alert data-user prospects to the fact that some systems are available on a lease–buy–option basis.

Among the important factors to consider before setting up any system are the volume of data, response time, accuracy (error rate), terminal locations, terminal types, and reliability (making the connection). Weighed against any of these factors is the economic criterion.

Urgency determines whether or not one should even use data telecommunications. After all, there are the postal service and private delivery systems. Often the data content of a magnetic tape can be flown to its destination more quickly than it can be transmitted by a voice-grade line. (One bank in the southeastern U.S. uses a helicopter to pick up data on magnetic tape from its branch offices.)

The choice between ARQ and FEC depends on such questions as the expected message length, availability of full-duplex lines, turnaround time, and the availability of buffers to store at the transmitting end a block of data in case it needs to be retransmitted. FEC is the only practical technique for error control when no reverse channel is available or justifiable.

Most data terminals are designed around a particular method of error control, and usually the user has no way of modifying it. In making his decision he must consider both the terminal equipment and modem as interacting subsystems.

Larger systems

Up to this point, only the simplest sort of system has been analyzed—namely, a terminal-to-terminal hookup.

A wide choice of network geometries is available; star (individual lines from the central to each terminal), multidrop (all terminals appear as drops off a single line running from the central to the most distant terminal), and loop (in which a single line leaves the central, and threads through all terminal points before returning to the central. By "central" is meant the main installation, or concentrator/multiplexor serving a group of terminals. Each geometry poses its problems in line control, error recovery, and economic tradeoffs. For example, in the multidrop situation, the entire multidrop net is tied up while a single transmission is in progress. Another geometry is the "multiconnected" one used where alternate routing is desired. Here, any one station may be linked to one or more of the other stations. These geometries are shown in Fig. 10.

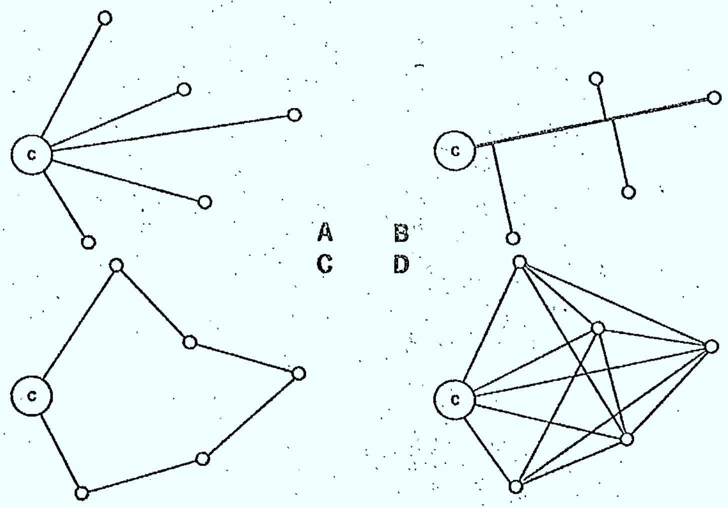
There are various options for switching and routing the traffic on its way from source to destination. In line switching, such as is used in the common-carrier voice plant, the initiation of communications between two points sets up a physical patch path that is maintained for the duration of the transaction. In message switching, each message is sent into the network before routing to the destination is determined. The message makes its way through the network with the destination address (given in the header preceding the message text) telling each station in the net where to forward the message. Since some circuits or stations may be busy, the message must often be stored at intermediate stations (thus leading to the term "store and forward" as an alternate name for this form of switching).

Each kind of switching has its advantages and disadvantages. Circuit switching currently requires connect times* (time to set up the connection) that are very long for data transmission (often tens of seconds), and ties up transmission capacity for long periods. However, the connection, once made, has a low end-to-end delay. Message switching makes more efficient use of lines by time-sharing them, provides quick connect time (so that the sending station can get rid of the message immediately), but may impose long end-to-end delays.

An example of a major system that makes use of mes-

*Computer people employ this term as used here, whereas in telephone company parlance, the term is used to designate the duration of the connection.

FIGURE 10. Network geometries include (A) star, (B) multidrop, (C) loop, and (D) multiconnected. Geographical locations are the same in each case and the central station is indicated by the letter "C."



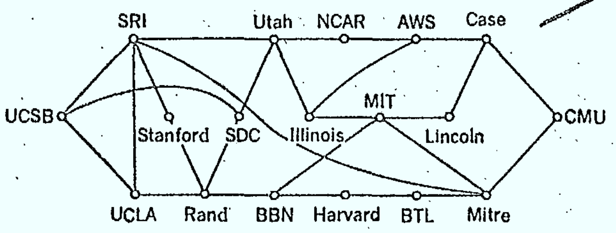


FIGURE 11. ARPA network with expanded topology at a cost of about $59 000 per node per year and a capacity of 23 kb per node.

sage switching and buffering over a multiconnected topology is that constructed by the Advanced Research Project Agency (ARPA). This system is intended to allow the use at one computing facility, for example, at a major university, of particular hardware and software capability unique to another remote such computing facility. The system (Fig. 11) uses 50-kb/s channels to interconnect computers in various universities and research institutions throughout the United States. Bell lines are used, but the switching gear, concentrators, store-and-forward equipment, small digital computers, and other components are ARPA-provided.

A likely system of the future resides with cable television (CATV). Presently, there are 2000 CATV station clusters (usually multidrop arrangements) serving some 4 million homes. Industry observers expect, however, that by 1980 CATV will have become a broadband system providing two-way communication over 20 to 40 channels. The potential of this network for data transmission largely has been untapped, but there has been considerable talk of possible data services that could be offered economically by the CATV industry, particularly for locally concentrated network applications.

Many data-communications experts will impress the potential customer with the fact that a large data-communications system cannot be put together and be fully operative within a matter of months or even a year. The system must be carefully thought out before the first piece of equipment is considered, and a complex system should be designed from the ground up if maximum utility is to be achieved. Consideration must be given to compromising between peak and average loads. Alternate, substitute, and temporary routings must enter into any scheme. It will usually profit the user to have computer simulations of possible system options. Many simulation languages exist, as well as several software packages for simulating large portions of an entire teleprocessing system.

A final thought about systems. It would be foolish to create a complex data-communications scheme based on a single type of communications channel. If the system depends on leased lines, perhaps DDD backup should be provided. This provision also will entail some thought about the capability of the equipment in the system to operate at a reduced rate or to cope with poorer channel characteristics without degraded results.

As for those users who are dissatisfied with today's communication facility service—for reasons of cost, connect time, or error rate—they may look forward to the digital networks now being planned and implemented.

We wish to thank the following for contributing to the material appearing in this article: National Electronics Conference (Professional Growth in Electronics Seminar), Paul Muench and Labe Southmayd of American Telephone and Telegraph Company, J. P. McCarthy of Sylvania Electronic Systems, Willard S. Jones of RFL Industries, Inc., and James E. Snodgrass and James R. Bookstaver of the IBM Corporation.

Particular thanks are extended to IEEE members Paul E. Green, Jr., and Robert W. Lucky for advice and counsel, without which this article would not have been possible.

REFERENCES

1. James, R. T., "Data transmission—the art of moving information," *IEEE Spectrum*, vol. 2, pp. 65–83, Jan. 1965.

2. Forney, D. G., Jr., "Coding and its applications in space communications," *IEEE Spectrum*, vol. 7, pp. 47–58, June 1970.

3. Martin, J. L., *Teleprocessing Network Organization.* Englewood Cliffs, N.J.: Prentice-Hall, 1970.

4. "Application for data transmission network system description," Data Transmission Company (submitted to the FCC Nov. 1969).

5. "MCI—a customized communications carrier," Microwave Communications of America (submitted to the FCC Sept. 8, 1969).

6. "Telecommunications modem survey," *Telecommunications*, vol. 4, pp. 12–20, Feb. 1970.

7. "Telecommunications acoustic coupler survey," *Telecommunications*, vol. 4, pp. 18–20, Oct. 1970.

8. "Telecommunications hard-copy survey," *Telecommunications*, vol. 4, pp. 16–20, Dec. 1970.

FURTHER READING

Some of the better books covering the broad picture of data communications are

Stein, S., and Jones, J. J., *Modern Communication Principles.* New York: McGraw-Hill, 1967.

Martin, J. L., *Telecommunications and the Computer.* Englewood Cliffs, N.J.: Prentice-Hall, 1969.

For details about the topics covered in the "Basics" section of this article, read

Benice, R. J., and Frey, A. H., Jr., "Comparison of error control techniques," *IEEE Trans. Communications Technology*, vol. COM-12, pp. 146–154, Dec. 1964.

Bennett, W. R., and Davey, J. R., *Data Transmission.* New York: McGraw-Hill, 1965.

Lathi, B. P., *Signals, Systems, and Communications.* New York: Wiley, 1965.

Lucky, R. W., Salz, J., and Weldon, E. J., *Principles of Data Communication.* New York: McGraw-Hill, 1965, Chap. 10–12.

Peterson, W. W., *Error Correcting Codes.* New York: Wiley, 1961.

In-depth information about existing carrier-facility capability can be garnered from Alexander, A. A., Gryb, R. M., and Nast, D. W., "Capability of the telephone network for data transmission," *Bell System Tech. J.*, vol. 39, pp. 431–476, May 1960.

Information about how minicomputers might fit into the scheme of things appears in Jurgen, R. K., "Minicomputer applications in the seventies," *IEEE Spectrum*, vol. 7, pp. 37–52, Aug. 1970.

The role of non-Bell System carriers such as Datran and MCI is discussed in Mathison, S. L., and Walker, S. A., *Computers and Telecommunications—Issues in Public Policy.* Englewood Cliffs, N.J.: Prentice-Hall, 1970.

Paul Hersch joined the IEEE Spectrum staff in June 1969. His previous editorial experience includes positions with Science and Technology, Plastics Technology, and Electronic News. Before moving over to the technical trade-publication field, he worked some ten years in industry, first as a mechanical engineer with the Research Division of the Curtiss-Wright Corporation and then as a physicist with the Nopco Chemical Company. Mr. Hersch received the B.Sc. degree from McGill University, Montreal, Que., Canada, in 1952 and the M.S. degree in physics from Stevens Institute of Technology, Hoboken, N.J., in 1960.

Carleton University
November 30,1971.

CODING THEORY & APPLICATIONS

Coding theory has a history no doubt unique among engineering disciplines; the ultimate theorems came first practical applications later. Between 1948 - when shannon first proposed his basic theorems on information theory - and the start of the space age, little practical application developed from the lessons of coding theory. In fact a standard feature at the IEEE conventions during this period was a session entitled "Progress in Information Theory", in which talks purporting to show that the theory was approaching practical application tended instead to confirm the prejudices of practical men that information theory would do nothing for them.

In retrospect there was two principal reasons for this lag. First, Shannon's coding theorems were existence theorems which showed that within a large class of coding schemes, these existed some schemes - nearly all, actually - that could give arbitrarily low error rates at any information rate up to an initial rate called channel capacity. The theorems gave no clue to the actual construction of such schemes, however, and the search for coding techniques capable of remotely approaching the actual capacity proved so difficult that a folk theorem was proposed: "All codes are good except those that we can find or think of".

The second problem was that, the channels of practical interest - telephone lines, cable, microwave, troposcalter, and HF radio proved not to have anything like the statistical regularity assumed in the proof of the coding theorems. In fact, most theorems are based on the assumption of statistical independence of the noise affecting each transmitted symbol, whereas on the channels just cited disturbances tend to be manifested in bursts spanning many bits. This is to say nothing of other anomalies that arise in practice.

Over the past decade, the situation has improved considerably. The problem of finding workable coding schemes has been recognized to be fundamentally a problem of finding decoders of reasonable complexity.

The solution has been sought in considering classes of codes so structured that efficient decoding becomes feasible (but not so much structured that the codes themselves are no good). The most popular approach has been to use the structures of abstract algebra to generate classes of good, decodable block codes. A second approach uses linear sequential circuits to generate a class of codes that are called convolutional; for most of the applications that the author is aware of, convolutional codes seem to have a better balance between structure and randomnes than is capable with the perhaps too structured block codes.

Basic Binary Codes

Suppose that we wish to transmit a sequence of binary digits occurs a noisy channel. Although we are unable to prevent the channel from causing errors, we can reduce their undesirable effects with the use of coding. The basic idea is simple - we take a set of k message digits which we wish to transmit, annex to them r check digits, and transmit the entire block of n $\equiv$ k+r channel digits. Assuming that the channel noise changes sufficiently few of these n transmitted digits, the r check digits may provide the receiver with sufficient information to enable him to detect and correct the channel errors.

Encoding Problem

Given any particular sequence of k message digits, the transmitter must have some rule for selecting the r check digits.

Information Rate $$R \equiv \frac{k}{n} = \underbrace{\frac{k}{k+r}}_{\text{block length}} = \frac{\text{message}}{\text{message} + \text{check}}$$

Code Word

Any n tuple i.e., sequence n which the encoder might transmit is called a code word.

NOTE Although they are 2^n different binary sequences of length n, only 2^k of these are code words, because the r check digits within any code word are completely determined by the k message digits.

n = 3 binary

2^3 = 8 different word

$$\left.\begin{array}{l}000\\001\\010\\011\\100\\101\\110\\111\end{array}\right\}\text{code words}$$

Code

The set consisting of 2^k codewords is called the code. The coding problem is given the n received digits, the decoder must attempt to decide which of the 2^k possible code words was transmitted.

Decoding Failure:

We commit a decoding failure when we cannot decode.

Decoding Error:

When we decode incorrectly.

We have that if $R = \frac{k}{n}$ is small then the block length is long and the probability of decoding error is very small. We are usually more interested in codes which have a high information rate.

Review of Algebra

Linear Codes

Mod 2 Arithematic

Addition

$+$	0	1
0	0	1
1	1	0

$\oplus$

Multiplication

X	0	1
0	0	0
1	0	1

$\otimes$

The modulo 2 sum of two n bit code words is defined as the bit by bit modulo 2 sum; that if x and y are code words then the bits in their sum are

$$z_i = x_i \oplus y_i \quad , \qquad i = 1,---,n,$$

Shift Register Sequences

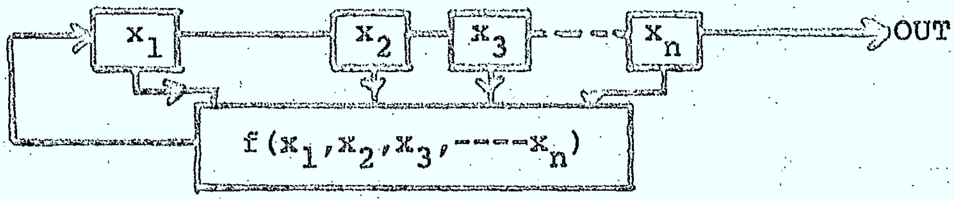


Figure 1 General Shift Register of Degree n with logical feedback

A shift register of degree n is a device consisting of n consecutive binary storage positions which shift the contents of each position to be next position down the line, in time to the regular beat of a clock (or other timing devices). In order to prevent the shift register from emptying by the end of n clock pulses, a "feedback term" may be computed as a logical function f (i.e., Boolean function), of the contents of the n positions and fed back into the first position of the shift register.

The maximum length shift register (or pseudo random or simplex) codes make a good introduction to algebraic block codes; their properties are interesting and easy to derive, and serve as an easy entree to the mysteries of finite fields, upon which further developments in block codes depend. (Remark - the number and quality of the pictures of returned from Mariner probes depend on the use of codes like these.

Example

Consider first a digital feedback circuit such as the one depicted in figure 2. That is a (m=4) bit shift register.

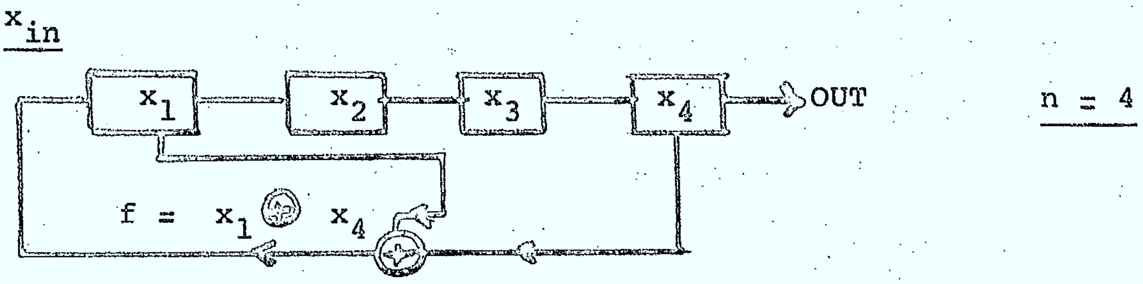


Figure 2

The input $x_{in} = x_1 \oplus x_4$; the state output

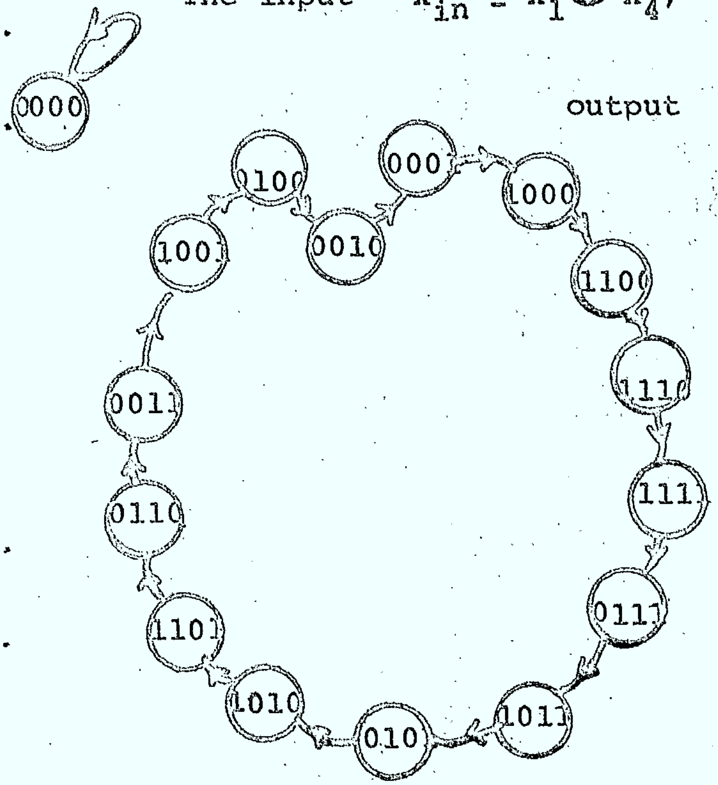


Figure 3

We find that fifteen shifts cycle the register through all non zero states and return the register to the starting state. The name maximum length linear shift register is given to this circuit since, given that 0000 must go to 0000, the fifteen state cycle is the maximum length possible.

Theorem

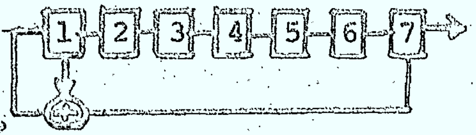
Even an SR of length R, the output sequence is always periodic with a period $p \leq 2^n$. If we have an LSR the period p is atmost $2^k - 1$, for $p = 2^k-1$. The shift register is called a Maximum length LSR.

Corrollary

It is a nontrivial result of algebra that for any number of stages n we can always find a circuit (shift register) like figure 2, with a state diagram like figure 3 and a period $p = 2^k- 1$. The input will always goes into the zero state on a shift. The remaining 2^k-1 form a max length cycle. The following table specifies input connections to the modulo 2 adders that will give a max. length shift register for $1 \leq k \leq 14$.

e.g. message k = 7

no. of stages = 7 → Register →



rate $R = \frac{7}{2^{k-1}} = \frac{7}{127}$

period p = 127

k	$(2^k-1,k)$	$\frac{k}{2^{k-1}}$	Stages connected to mod 2 adder
1	(1,1)		2
2	(3,2)		1,2
3	(7,3)		1,3
4	(15,4)		1,4
5	(31,5)		1,4
6	(63,6)		1,6
7	(127,7)		1,7
8	(225,8)		1,5,6,7
9	(511,7)		1,6
10	(1023,10)		1,8
11	(2047,11)		1,10
12	(4097,12)		1,7,9,12
13	(9145,13)		1,10,11,13
14	(18391,14)		1,5,9,4
34			1,8,33,34

NOTE:

The information rate becomes small as k is large; which limits the usefulness of these codes for coding purposes: in other applications, however, the fact that a very long nonrepeating sequence can be generated with a shift register is the feature of interest.

A block code using the circuit of figure 2 as an encoder operates as follows: The message to be transmitted which is assumed to be a sequence of bits, is segregated into 4 bit segments. Each segment is loaded into the 4 bit shift register and the register is shifted 15 times. The fifteen bits coming out of the right most stage of the register are transmitted as a block or code word. Table 2 gives the 15 bit code words corresponding to each 4 bit information segment.

Table 2 (look at figure 3) Code words in (15,4) code from MLSR

Message Bits	Code Words
0000	000000000000000
0001	000111101011001
1000	100011110101100
0100	010001111010110
0010	001000111101011
1001	.
1101	.
0110	.
1011	etc.
0101	.
1010	.
1101	.
1110	.
1111	.
0111	.
0011	.

Properties of Sequences generated by Maximum Length Linear Shift Registers

Given a 2^k-1 bit sequence

(a) First the bits in this sequence are the right-most output bits of the 2^k-1 non zero state sequences of length k.

(b) Since exactly half of all k bit sequences end in 1, precisely $\frac{2^k}{2}$ ones occur in any maximum length sequence (for example 8 ones out of 15 bits in figure 3). In any period of the sequence the number of ones differs from the number of zeros by at most 1. (Called Balance property).

(c) In a long sequence, if we look at the output at a random time the probability of "seeing" a 1, is

$$\frac{\frac{2^k}{2}}{2^k-1} \approx 1/2.$$

Furthermore since all k bit sequences except the all zero sequence, occur somewhere in the maximum length sequence, the probability of seeing a one given any k-1 or fewer preceeding bits is still nearly 1/2. These and other statistical properties below, make a maximum length sequence difficult to distinguish from a sequence generated truly randomly, as by flipping a coin, yet these sequences are easy to generate and are repeatable. Thus they are commonly used to generate pseudo random bits. They are called pseudo random sequences.

(d) (Run Property) Among the ones and zeros in each period one half the owns of each kind are of length one, one fourth of each kind are of length two, one eighth are of length three, etc. as long as these fractions give meaningful number of runs.

(e) (Correlation Property) If a period of the sequence is compared term by term, with any cycle shift of itself, the number of agreement differs from the number of disagreements by at most 1.

Using figure (3) to illustrate (e), consider the code word, outlined

```
000111101011001    } Number of agreements = 7
001111010110010    }
_______________
001000111101011      Note this is again a displacement of
                     one of the code words.
```

called parity check sequence.

__Theorem__

for all max length shift register sequences, its modulo 2 sum with any non trivial displacement of itself, gives a parity check sequence which is again a displacement of one of the code words.

__Corrollory__

If we form the modulo 2 sum of any two code words we get another code word. This is a group property of shift register codes and gives immediate answers to questions about distance or correlation between code words.

Definition

Hamming distance: between two code words is defined as the number of places in which the two words differ.

NOTE:

$$\text{distance }(a,b) = \#\text{ of ones }(a \oplus b) \text{ mod 2 Sum}$$

If we form the modulo 2 sum, we have 1 in the position in which they disagree. The # of these is the distance.

Theorem

The distance between any two code words $= \frac{2^k}{2}$ which is exactly the number of ones in their sum which is the same as the number of ones in any of the code words.

$\frac{16}{2} = 8$ in our example.

Proof

From the group property, their sum is another code word, but we had that the # of ones is $\frac{2^k}{2}$ in any code word, ∴ Number of ones and distance under modulo 2 sum coincide. This the distance between any two code words in these codes is $\frac{2^k}{2}$ or about half the code length.

This equidistant property of maximum length shift register codes makes them an optimum solution to the following problem in Signal design. "How can one construct 2^k equal energy signals, to minimize the cross correlation between any two signals, with no bandwith limitations"?

Let us suppose that a code word is sent by P.S.K. so that a 0 is sent as a band of amplitude -1 and 1 as an amplitude of + 1. The 2^k code words then correspond to 2^k vectors in 2^k-1 dimensions, all of equal energy (<u>auto correlation</u>). The <u>cross correlations</u> (inner products) of any two vectors is a sum of baud by baud correlation equal to + 1 if they agree and -1 if they disagree.

But we have just proved that the Hamming distance between any two code words is $\frac{2^k}{2}$, so that the vectors disagree in $\frac{2^k}{2}$ places, and ∴ agree in the remaining $\frac{2^k}{2} - 1$.

Consequently any two vectors are anti-correlated with cross correlation -1. This implies that as vectors in $(\frac{2^k}{2} - 1)$ space, the code words form a geometrical object called a simplex, which is universally believed (though it has never been quite proven) to be the distribution of equal energy signals in signal space that minimizes the probability of incorrect detection. Figure (4) shows the simplex corresponding to the k = 2 maximum length shift register code, which takes the form of a tetrahedron in 3 dimensions. Here is an intriguing contact between algebraic coding theory and the geometry of N dimensions.

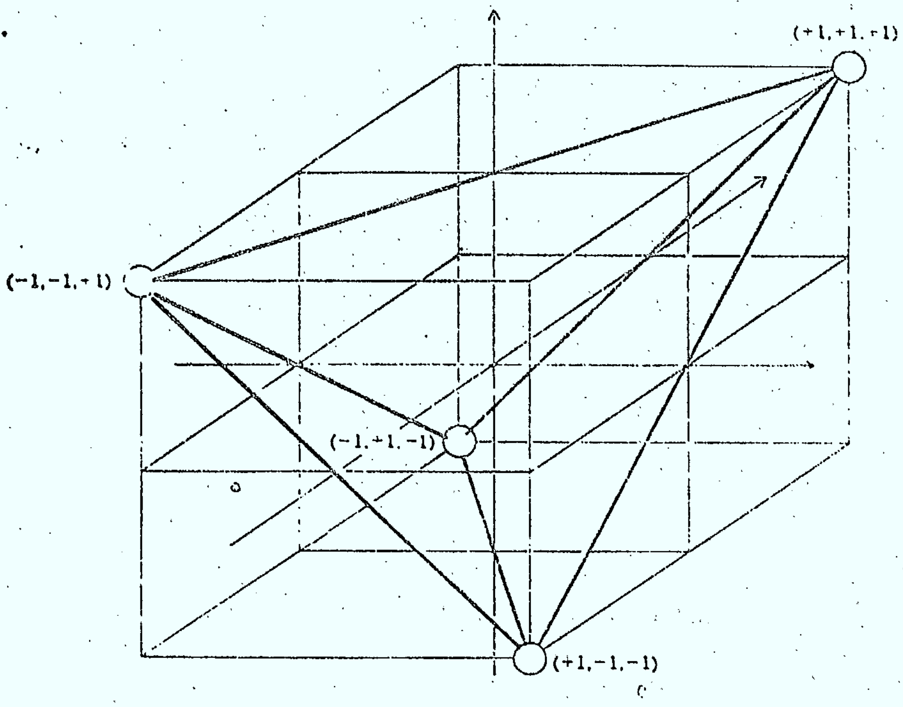


Figure 4

Simplex (tetrahedron) formed by k = 2, (3,2) code in three dimensions.

REFERENCES

1. Digital Communications with Space Applications by S.W. Golomb, Prentice-Hall, Inc. 1964.

2. See Also Paper on Coding Theory by David Fourney, IEEE Spectrum, Spring 1970.

Course 99-596
Carleton University

December 7, 1971

Dr. John deMercado

Errors Their Treatment and Error Correcting Codes

In these lectures we shall consider the treatment of errors arising from noise and distortion on communication lines.

The following Table 1, gives typical average error rates, and you can use them as a rule of thumb figures for doing probability calculations. With them, it will be possible, as we shall see, to answer such questions as what degree of error checking and what block length should be used for transmission for different systems.

Average Type of Channel	Transmission Rates (bits/sec.)	Bit error rate 1 in.
50 baud telex	50	50,000
150 or 250 baud subvoice grade lines	150 or 200	100,000
Public voice lines	600 1200 *)2400	500,000 200,000 100,000 (10^5)

Possible Approach

(1) Simply ignore the noise, in fact, the majority of telegraph links in operation today, for example, have no error checking facilities at all. Part of the reason of course is,

*) It is likely that transmission over a voice line at 4800 bits/sec. will give higher error rate than those above.

that they normally transmit english language text, and errors caused by the ranging of a bit or small group of bits usually are almost always automatically corrected by the brain. If the text is unintelligible, then the receiver can always ask for a re-transmission.

A second reasons, is that one bit in 10^5 errors is not quite as bad as it sounds. For example James Martin estimated that his book of about 300 pages and containing about 110,000 words, after being edited word by word by team of professional edition, still had an error rate of 1 bit in 2×10^4 bits using a five bit code. Taking each word to be 8 letters and each letter requiring 5 bits we need $5 \times 8 \times 10^5$ bits to code the book, if we then transmitted over a line with error rate 1 in 10^5 about 40 of these letters would be wrong. Thus the error rate of 1×10^5 on an nocheck transmission line is better than the carefully proof read text.

On the other extreme, a coding scheme is on the market which gives an undetected error rate of 1 bit in 10^{14}, but this is expensive; but let us estimate what it means to have 1 in 10^{14}.

Suppose we transmit at 2400 bits/sec. for 2000 years since the time of (Chirst), we would have transmitted ($\approx 2 \times 10^3 < 10^{14}$) bits and probably not have had an error yet

$$\begin{aligned} & 2400 \times 60 \times 60 \times 365 \times 2000 \\ \approx\ & 2.4 \times 10^{11} \times 3.6^2 \times 2 \\ \approx\ & 2 \times 10^{13} < 10^{14} \end{aligned}$$

Criteria for Choice of A Code

The merit of any scheme for correcting transmission errors is a function of three properties.

(1) What is its efficiency in detecting errors? How many incorrect messages does it let through? Ideally, we would like a scheme which catches all errors.

(2) How much does it reduce the line throughput? Both redundant bits and retr ansmission lessen the total data throughput on the line.

(3) How much does it cost?

In the early days, (3) dictated that low accuracy was often taken in favour of high costs. However, with the advent of high speed modems and low cost large scale integrated circuits the balance is now swinging in favour of more complex codes.

Error Correcting Codes

The ability of a code to correct errors is related to its ability to detect them. For instance a code that can detect double errors can correct single ones. In general a a device that can detect 2x errors can correct x. Similarily, some codes can detect two error burts of length b bits; such codes could correct one such burst.

In data transmission a single noise pulse or drop out (loss of signal) is frequently of greater duration than the length of one bit. This is more likely to be so when a high bit rate is used. Even at low bit rates double errors are common. A CCITT*) study of 50 band telegraph lines give the following figures:-

Isolated single bit errors	50-60%
Error bursts with two erroneous bits	10-20%
Error bursts with three erroneous bits	3-10%
Error bursts with four erroneous bits	2-6%

A burst was defined here as bits in error separated by less than ten non-erroneous bits.

*) CCITT Special Study Group "Date Transmission" Contribution 92, Annex XIII p. 131 (Oct. 18, 1963)

Polynominal Codes *)

All codes, such as M out of N codes, Hamming codes, Bose-Chaudri Codes, The Fire Codes, the codes of Milas, etc.) can be described in terms of the divisions of polynomials. Polynomial Codes can be made to perform with very high efficiency.

Let us suppose the data block to be transmitted has k bits, we can represent this as a polynomial in a variable x having k terms. i.e., a polymial of order k-1, for example, suppose the message being is 1010001101, the polynomial.

$$M(x) = x^9 + x^7 + x^3 + x^2 + 1$$

in general word $[a_{k-1}, ---, a_o]$ can be represented by

$$M(x) = a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + --- + a_1 x + a_o$$

Where the high order term of the polynomial is the bit that is transmitted first. That is we transmit from "right to left"

Addition in Modulo 2 $\oplus$

$\oplus$	0	1
0	0	1
1	1	0

Multiplication Modulo 2 $\otimes$

$\otimes$	0	1
0	0	0
1	0	1

Now there is a convenient way of expressing the messages to be sent. We will manipulate them using the laws of ordinary algebra, except that modulo 2 addition must be employed. We illustrate this as,

*) "Cyclic Codes for Error Detection" by W.W. Peterson & D.T. Brown, proceedings of the IRE, January 1961

Mod 2 $\oplus$ of two polynomial is

$$\begin{array}{l} x^7 + x^6 + x^5 + -- + x^2 + 1 \\ x^7 + \quad\quad + x^5 + x^4 + x^3 + x^2 + \\ \hline \quad x^6 + \quad\quad + x^4 + x^3 + \quad\quad 1 \end{array} \left.\vphantom{\begin{array}{l}1\\1\\1\end{array}}\right\} \quad \begin{array}{c} 1\;1\;1\;0\;0\;1\;0\;1 \\ 1\;0\;1\;1\;1\;1\;0\;0 \\ \hline 0\;1\;0\;1\;1\;0\;0\;1 \end{array}$$

Multiplication Mod 2 $\otimes$

$$\begin{array}{l} (x^7 + x^6 + x^5 + x^2 + 1)\ (x + 1) \\ = \\ x^8 + x^7 + x^6 + x^3 + x \\ \quad\quad x^7 + x^6 + x^5 + x^2 + 1 \\ \hline \underbrace{x^8 + -- \quad x^5 + x^3 + x^2 + x + 1}_{1\;0\;0\;1\;0\;1\;1\;1} \end{array} \left.\vphantom{\begin{array}{l}1\\1\\1\\1\\1\end{array}}\right\} \quad \begin{array}{r} 1\;1\;1\;0\;0\;1\;0\;1 \quad X\;1\;1 \\ = \quad\quad\quad\quad\quad\quad \\ 1\;1\;1\;0\;0\;1\;0\;1\;0 \\ 1\;1\;1\;0\;0\;1\;0\;1 \\ \hline 1\;0\;0\;1\;0\;1\;1\;1\;1 \end{array}$$

To transmit the k data block M(x) we need a second polynomial referred to as the generating polynomial P(n), of degrees r, where $k \geqslant r \geqslant 0$. P(x) has unit coefficient on the x^o, ie, the lowest term is 1.

For example To transmit the message

$$M(x) = x^9 + x^7 + x^3 + x^2 + 1$$

we might use a generating polynominal

$$P\ (x) = x^5 + x^4 + x^2 + 1$$

the steps involved in the transmission are in effect as follows.

Step 1 The data message $M(x)$ is multiplied by x^r, giving r zeros (o's) in the low order positions.

Step 2 The result is divided by P(x) , this gives a unique quotient Q(x) and remainder R(n).

$$\frac{x^r \; M(x)}{P(x)} = Q(x) \oplus \frac{R(x)}{P(x)}$$

Step 3 The remainder is added to the message thus placing up to r tenms in the r lower orders positions, this is the message that is transmitted. Let us call it T(x), then $T(x) = x^r \; M(x) \oplus R(n)$

Suppose $M(x) = x^9 + x^7 + x^3 + x^2 + 1 \equiv 1\,0\,1\,0\,0\,0\,1\,1\,0\,1$

and $P(x) = x^5 + x^4 + x^2 + 1 \equiv 1\,1\,0\,1\,0\,1$

Step 1 $x^r \; M(x) = x^5 \; M(x) = x^{14} + x^{12} + x^8 + x^7 + x^5 =$

1 0 1 0 0 0 1 1 0 10 0 0 0 0

Step 2 $$\frac{x^5 \; M(x)}{P(x)} = \frac{x^{14} + x^{12} + x^8 + x^7 + x^5}{x^5 + x^4 + x^2 + 1}$$

$$\overbrace{x^9 + x^8 + x^6 + x^4 + x^2}^{Q(x)} + x,$$

$$\begin{array}{lrl}
x^5 + x^4 + x^2 + 1 & \big) & x^{14} + x^{12} + x^8 + x^7 + x^5 \\
 & & \underline{x^{14} + x^{13} + x^{11} + x^9} \\
 & & x^{13} + x^{12} + x^{11} + x^9 + x^8 + x^7 + x^5 \\
 & & \underline{x^{13} + x^{12} + x^{10} + x^8} \\
 & & \quad x^{11} + x^{10} + x^9 + x^7 + x^5 \\
 & & \quad \underline{x^{11} + x^{10} + x^8 + x^6} \\
 & & \qquad x^9 + x^8 + x^7 + x^6 + x^5 \\
 & & \qquad \underline{x^9 + x^8 + x^6 + x^4} \\
 & & \qquad\quad x^7 + x^5 + x^4 \\
 & & \qquad\quad \underline{x^7 + x^6 + x^4 + x^2} \\
 & & \qquad\qquad x^6 + x^5 + x^2 \\
 & & \qquad\qquad \underline{x^6 + x^5 + x^3 + x} \\
 & & \qquad R(x) \Longrightarrow x^3 + x^2 + x
\end{array}$$

$$\boxed{\begin{array}{l} Q(x) = 1101010110 \\ R(x) = 1110 \end{array}}$$

(3) The remainder R(x) is added to x^r M(x), to give T(x) the message transmitted

$$\begin{array}{lrl}
\therefore \; T(x) & 1\,0\,1\,0\,0\,0\,1\,1\,0\,1\,0\,0\,0\,0\,0 & \equiv M(x) \\
 & \oplus \qquad\qquad 1\,1\,1\,0 & \equiv R(x) \\
\hline
 & \underbrace{1\,0\,1\,0\,0\,0\,1\,1\,0\,1}_{\text{We are thus sending the original bit pattern, with five bits accompanying it for error detection.}}\underbrace{0\,1\,1\,1\,0}_{\text{Check Bits}} &
\end{array}$$

The question remains, what is this all in aid of, let us return to

$$\frac{x^r\ M(x)}{P(n)} = Q(n) \oplus \frac{R(x)}{P(x)}$$

we have rewriting this

$$x^r\ M(x) = Q(x)\ P(x) \oplus R(x)$$

since substraction and addition in modulo 2 or the same, we can rewrite the above equation as

$$x^r\ M(x) + R(x) = Q\ (x)\ P(x)$$

or

$$T(x) = Q\ (x)\ P(x)$$

"The message transmitted is therefore exactly divisible by the generating Polynomial P(x).

It is this property that we check in attempting to see whether an error has occurred. The receiving machine, in effect divides the message polynomial T(x) by P(x), if the remainder is non zero, then an error has occurred. If it is zero, then either there is no error or an undectable error has occurred.

To see that to speak of an undetable error makes sense, consider - the pattern of error bits, E(x), ie, bits of T(x) that are changed by noise in the transmitted message. Thus we receive $T(x) + E(x)$

Thus if $T(x) + E(x)$ is exactly divisible by P(x), then the remainder is zero and we cannot detect. On the other hand if E(x) is not divisible by P(x) we will detect it Knowing the characteristics of the communication lines, we must, therefore pick the generating polynomial $P(x)$ so that the pattern of error bits $E(x)$ will not be divisible by it.

Error Detection Probabilities

The choice of generating polynomial should be dependent on a knowledge of the error patterns that are likely to occur on the channel in question. There are certain error characteristics that we can be sure to protect the data from.

Case 1 - Single Bit Errors

If the message block $M(x)$ protected by the polynomial code $T(x) = M(x) \oplus R(x)$ has one single bit in error then $E(x) = x^i$, where is is less than the total number of bits in the message $T(x)$. (let us say $n = r+k$).

$\therefore$ if we give our generating polynomial $P(x)$ more than one term, then x^i cannot be divided by it exactly then all single bit errors will be detected.

Case 2 - Double Bit Errors

Double bit errors can be represented by the polynomial $E(x) = x^i + x^j$ where i and j are both less than the number of bits in $T(x)$, ie, less than n. Let us say $i \leqslant j$, then we can write $E(x) = x^i \ (1 + x^{j-1})$. Therefore the error to be detected, neither $x^i \ (1 + x^{j-1})$ may be divisible by the generating polynomial. Thus if this polynomial $P(x)$ has a factor with three terms, then this will be so and all double errors can be detected.

Case 3 - Odd Numbers of Errors

Theorem If the error message contains an odd number of bits in error, then the polynomial $T(x)$ that represents it is not divisible by $(x + 1) = P(x)$

Proof Suppose that the message is represented by a polynomial $T(x)$ which is divisible by $(x + 1)$ then we have

$$E(x) = (x+1)\,Q(x)$$

put $x = 1$ we get

$$E(1) = \underbrace{(1+1)}_{0}\,Q(x)$$

$$\therefore E(1) = 0$$

Therefore $E(x)$ must contain an even number of terms, hence if we employ a generating polynomial $P(x)$ with a factor $(x + 1)$, then any message with an odd number of errors will be caught.

Comments

Now any polynomial of the form (x^c+1) contains a factor $(x + 1)$ since $(x^c + 1) = (x + 1)\,(x^{c-1} + x^{c-2} + --- + 1)$. Therefore any generating polynomial of the form $(x^c + 1)$ will detect all errors with an odd number of bits incorrect.

Case 4 Bursts of Errors

Defn. a burst of length b is defined as the number of bits in a group having at least its first and last bits in error. Thus if $E(x)$ represents the error pattern

$$\underbrace{0\;0\;0\;0\;0\;1\;0\;1\;0\;0\;1\;1\;0\;0\;0\;0}_{b\;=\;7}$$

this contains a burst of length $b = 7$

$\therefore$ $E(x) = x^{10} + x^8 + x^5 + x^4$

now we can always factor $E(x) = x^i\ E(x)$ where i is less than the number of bits in the message for example.

$$E(x) = x^4\ (x^6 + x^4 + x + 1)$$

Now in general x^i is not divisible by a $P(x)$ containing only one term. Therefore the burst error will go undetectable off $E_1(x)$ is exactly divisible by $P(x)\ \big((x+1)\big)$. Now when the length of the burst b is less than the length $r + 1$ of $P(x)$, the generating Polynomial $E_1(x)$ can be detected. Thus if we use a generating polynomial of 13 bits, all bursts of length 12 bits or less will be detected. (recall $T(x) = x^r\ M(x) + R(x)$ $\therefore$ to achieve this we will have to use $r = 12$ redundant bits in the message, $\therefore$ $R(x) = 12$ bits

Case 2 $b \geqq r + 1$

Now if $b = r+1$ that is the number of bits in the burst = number of bits in the generating polynomial, then the error will go undetected if the burst $E(x)$ is identical to the generating polynomial $P(x)$. Since the first and last bits in the burst are by definition error bits, therefore the remaining $(r-1)$ bits must correspond to those of the generating polynomial. If we regard all combinations of bits as equally probable then the probability if find a specific pattern of $(r-1)$ bits we be the probability that $(r-1)$ independent bits are identical with that of the generating polynomial.

This probability is $\left(\frac{1}{2}\right)^{r-1}$ for $r = 12 \approx .00049$. So that the probability of an undetected error is a very rare event given that a burst $b = r+1 = 13$ bits occurs.

Case 3 $b > r+1$

When the number of bits in the burst b is greater than (r+1), there is a variety of possible error patterns that are divisible by P(x). (The question is how many). If $E_1(x)$ is divisible by P(x) then we can write

$$\underbrace{E_1(x) =}_{\text{degree } b-1} \underbrace{Q_1(x)}_{\text{degree } b-1-r} \underbrace{P(x)}_{r}$$

say $E_1(x)$ is a polynomial of degree (b-1) containing b terms, p(x) is a polynomial of degree r and contains r+1; terms in the degree of the polynomial $Q_1(x) = b-1-r$, $\therefore$ the number of bits represented by $Q_1(x) = b-r$. The first and last terms of $E_1(x)$ are always 1, this causes the first and last terms of $Q_1(x)$ to always be 1. These are therefore b-r-2 terms in $Q_1(x)$ which can alternate in value. This means that there are b-r-2 terms in $Q_1(x)$ which can alternate in value. This means that these are b-r-2 ways in which P(x) can divide $E_1(x)$.

Now $E_1(x)$ can have $2^{(b-2)}$ possible combinations give the first and last are fixed, $\therefore$ the probability of an error being detected, ie, the probability that P(x) divides $E_1(x)$ is

$$\frac{2^{b-2-r}}{2^{b-2}} = 2^{-r}$$

Comment

On the above example in which r=12, the probability of an undetected error is $2^{-12} \hat{=} .00024$, given that the burst contains a length greater than 13 bits (again a rare occurrence).

To Summarize

If we choose a polynomial P(x) having (x+1) as a factor and one factor with three or more terms, then the following protection will be given

Single bit errors	: 100% protections
Two bits in error (separate or not)	: 100% protection
An odd number of bits in error	: 100% protection
An error burst of length less than (r+1) bits	: 100% protection
An error burst of exactly (r+1) bits in length	: $\left(1-(\frac{1}{2})^{r-1}\right)$ probability of detection
An error burst of length greater than (r+1) bits	: $\left(1-(\frac{1}{2})^{r}\right)$ probability of detection

Assume an equal probability for bits in an error pattern

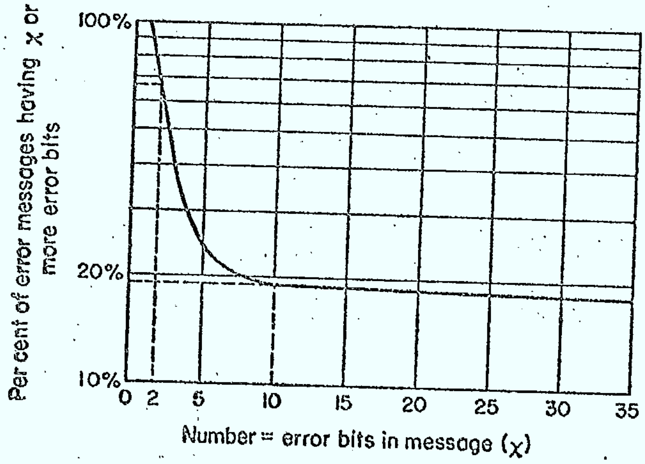


Figure 1

Explanation of Fig. 1

Figure 1, shows measurements of burst length mode on a long distance leased line in Europe. In these measurements the data was sent in block T(x) of 792 bits; of these blocks, 36 percent of those in error had only one bit in error, 81% had less than 10 bits, but 15 percent had large numbers of errors.

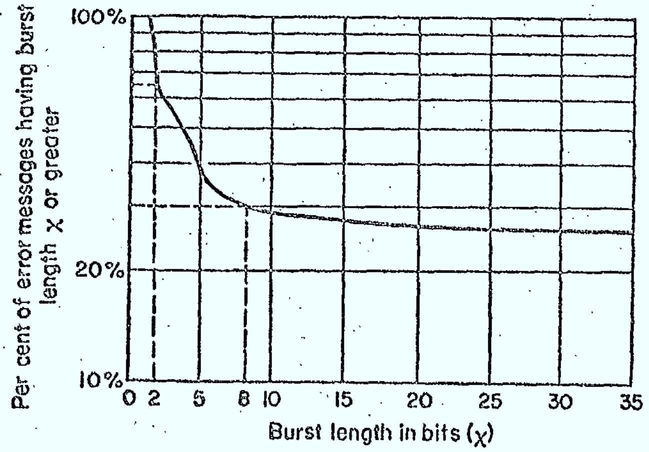


Fig. 5.3. Burst lengths and numbers of error bits per error message encountered on data transmission tests, at a 2000 bit/sec transmission rate, on a multipoint leased voice line from London to Rome.* Fixed message lengths of 792 bits were used. *_Data Transmission Test on a Multipoint Telephone Network in Europe, CCITT Blue Book_, _Supplement No. 37. Published by the International Telecommunication Union, Geneva, November 1964._

Figure 2

Comes from "Data Transmission Text on a Multipoint Telephone Network In Europe", CCITT blue book, Supplement #37 published by ITU Geneva Nov. 1964

From the above figure 2, we see that there is a fairly high proportion of long bursts , ie, 35 bits in length. This means that some errors are not going to be caught by the polynomial checks, or for that matter by any other checking schemes that are reasonable to implement.

Explanation of Figure 2

Figure 2 shows the distribution of burst lengths. A burst here is defined as the distance in bits between the first bit error and last bit error in the blocks. This curve states that 36 percent of the error blocks have a burst length of only one bit, 34% have 2 - 8 bit bursts, 30% have greater than 8 bits

Results Obtained In Practice

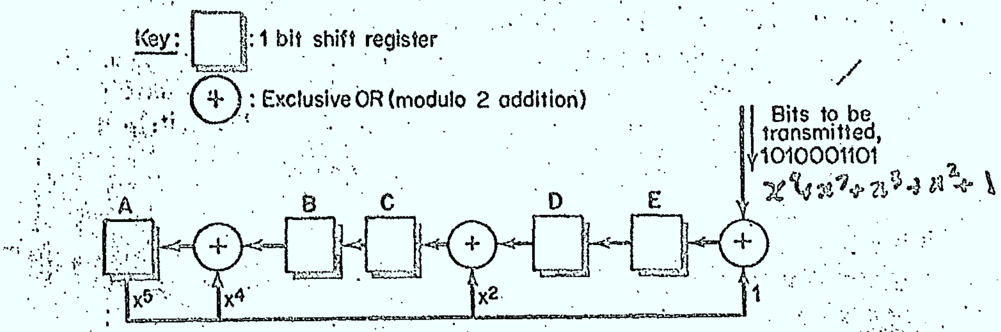
Results obtained on practive with polynomial codes on telephone lines have been reasonably close to the theoretical prediction. Figure 3 shows a typical set of results. These measurements were made on a leased line from London to Rome transmitting random bit messages in blocks of 729 bits. The transmission speed was 2000 bps.

Generating Polynomial Used	Fraction of	Undetected Error Message
	Expected Predicted	Actual
$x^6 + 1$	.0156	.0107
$x^6 + x^5 + 1$	.0156	.0075
$x^6 + x + 1$	.0156	.0066
$x^7 + x^3 + 1$	.0078	.0035
$x^{12} + 1$	.0003	.0021
$x^{12} + x^{11} + 1$	.0003	.0021
$x^{12} + x + 1$	.0003	

Figure 3

Encoding and Decoding Circuits

Polynomial codes are quite easy to code and decode; the division of $\frac{x^r M(x)}{P(x)} = Q(x) + R(x)$, to get R(x) can be performed with a series of 1 bit shift register and modulo 2 adders (exclusive or circuits). The number of shift register positions is the same as the degree of the divisions - five for the division in figure . The number of exclusive or circuits is equal to the number of 1 bits in the divisor - 1 = 3 for figure 4 .



Contents of shift registers:

	A	B	C	D	E	Input bit	
Initial contents:	0	0	0	0	0		
Step 1	0	0	0	0	1	1	Message to be sent
Step 2	0	0	0	1	0	0	
Step 3	0	0	1	0	1	1	
Step 4	0	1	0	1	0	0	
Step 5	1	0	1	0	0	0	
Step 6	1	1	1	0	1	0	
Step 7	0	1	1	1	0	1	
Step 8	1	1	1	0	1	1	
Step 9	0	1	1	1	1	0	
Step 10	1	1	1	1	1	1	
Step 11	0	1	0	1	1	0	Five 0's added
Step 12	1	0	1	1	0	0	
Step 13	1	1	0	0	1	0	
Step 14	0	0	1	1	1	0	
Step 15	0	1	1	1	0	0	

Remainder (which is sent as the five check bits)

Figure 4

Circuit with shift registers for dividing by the polynomial $(x^5 + x^4 + x^2 + 1)$.

The following figure 4, shows the circuit which performs the $\frac{x^r M(x)}{P(x)}$, where r=5, M(x) = 1010001101 and leaves the remainder R(x) = 1110 in the registers, similarily if $x^r M(x) + R(x)$ was entered the circuit would divide it, by P(x) and if the registers were empty, there would be no remainder, ie a bunch of zeros, indicating no remainder.

Reference

J. Martin - Teleprocessing Network Organization
Prentice Hall, 1970.

The conventional solution to the problem is shown in Fig. 10. A few moments of study will indicate that it does in fact perform the required operation. The Flow Table Logic solution is shown in Fig. 11. Again the problem is easily checked for correct performance.

The simplicity of the wiring and the regularity of the circuit are quite apparent. Such an approach should certainly prove valuable when batch-fabricated devices become a reaility. As was stated earlier, when applied to one such technology (EL-PC) the design and fabrication of circuits is greatly simplified. An early model of an EL-PC combination lock is shown in Fig. 12.

Conclusion

The Flow Table Logic technique for circuit design presented here was intended for use with batch-fabricated (or perhaps micro-miniature) devices, hence the emphasis on simplicity and regularity. These are obtained in some cases at the expense of actual component count. The exchange was felt to be acceptable, however, since minimizing the number of active elements is not guarantee of minimum cost. An interesting point that should be considered is the logical delay associated with circuits designed by the Flow Table Logic technique. The circuit can go from one state to any other state in approximately two logical delays. Thus one has not sacrificed speed in the quest for regularity. The ease of circuit design should also be an advantage of this technique.

As is true in most developments, there are some problem areas that still require investigation. The coding of the input lines and, in fact, the coding of the states of the flow table are far from optimum. Only further work can reveal whether this can be improved without sacrificing the simplicity of the circuit. In addition, the necessary delay required is not found in all technologies and thus must be carefully considered in the solution of a problem.

Acknowledgment

The material presented represents the efforts of the authors plus that of R. J. Domenico. In addition, the authors are indebted to E. J. Skiko, J. Earle, and R. Robeien for many helpful discussions. The EL-PC combination lock was constructed by Dr. J. A. O'Connell and B. Narken.

2107
54

for comm
EE5654-

Cyclic Codes for Error Detection*

W. W. PETERSON†, MEMBER, IRE, AND D. T. BROWN‡, MEMBER, IRE

Summary—Cyclic codes are defined and described from a new viewpoint involving polynomials. The basic properties of Hamming and Fire codes are derived. The potentialities of these codes for error detection and the equipment required for implementing error detection systems using cyclic codes are described in detail.

Introduction

Of THE many developments in the area of error-detection and error-correcting codes during the past three years, probably the most important have pertained to cyclic codes. Since their introduction by Prange,[1] very attractive burst-error correcting cyclic codes have been found by Abramson,[2,3] Fire,[4] Melas,[5] and Reiger.[6] Cyclic codes for correcting random errors have been found by Prange,[1] Green and San Soucie,[7,8] Bose and Ray-Chaudhuri,[9] and Melas.[10] Encoding and

* Received by the IRE, August 1, 1960; revised manuscript received October 28, 1960.

† University of Florida, Gainesville, Fla.

‡ IBM Corp., Poughkeepsie, N. Y.

[1] E. Prange, "Cyclic Error-Correcting Codes in Two Symbols," Air Force Cambridge Research Center, Bedford, Mass., Tech. Note AFCRC-TN-57-103, September, 1957; "Some Cyclic Error-Correcting Codes with Simple Decoding Algorithms," Tech. Note AFCRC-TN-58-156, April, 1958; "The Role of Coset Equivalence in the Analysis and Decoding of Group Codes," Tech. Note AFCRC-TR-59-164; June, 1959.

[2] N. M. Abramson, "A Class of Systematic Codes for Non-Independent Errors," Electronics Res. Lab., Stanford University, Stanford, Calif., Tech. Rept. No. 51; December, 1958.

[3] N. M. Abramson, "Error Correcting Codes from Linear Sequential Networks," presented at the Fourth London Symp. on Information Theory, London, Eng.; August, 1960.

[4] P. Fire, "A Class of Multiple-Error-Correcting Binary Codes for Non-Independent Errors," Sylvania Electric Products, Inc., Mountain View, Calif., Rept. No. RSL-E-2; March, 1959.

[5] C. M. Melas, "A new group of codes for correction for dependent errors in data transmission," *IBM J. Res. Dev.*, vol. 4, pp. 58–65; January, 1960.

[6] S. H. Reiger, "Codes for the correction of clustered errors," IRE TRANS. ON INFORMATION THEORY, vol. IT-6, pp. 16–21; March, 1960.

[7] J. H. Green, Jr., and R. L. San Soucie, "An error-correcting encoder and decoder of high efficiency," PROC. IRE, vol. 46, pp. 1744–1755; October, 1958.

[8] N. Zeiler, "On a Variation of the First Order Reed-Muller Codes," M.I.T. Lincoln Lab., Lexington, Mass., pp. 34–80; October, 1958.

[9] R. C. Bose and D. K. Ray-Chaudhuri, "A class of error-correcting binary group codes," *Information and Control*, vol. 3, pp. 68–79, March, 1960; "Further results on error correcting binary group codes," *Information and Control*; to be published.

[10] C. M. Melas, "A cyclic code for double error correction," *IBM J. Res. Dev.*, vol. 4, pp. 364–366; July, 1960.

error correcting procedures for these codes are relatively easily implemented using shift-registers with feedback connections.[11,12]

The first function of this paper is to introduce cyclic codes from a new viewpoint requiring only elementary mathematics and to derive the basic properties of Hamming and Fire codes. Second, the potentialities of cyclic codes for error detection and the equipment required for implementing error detection systems using cyclic codes are described in detail.

Polynomial Representation of Binary Information

We will be concerned with coding a message of k binary digits by appending $n-k$ binary digits as a check and transmitting the k information digits and then the $n-k$ check digits. It is convenient to think of the binary digits as coefficients of a polynomial in the dummy variable X. For example, a message 110101 is represented by the polynomial $1+X+X^3+X^5$. The polynomial is written low-order-to-high-order because these polynomials will be transmitted serially, high-order first, and it is conventional to indicate signal flow as occurring from left to right.

These polynomials will be treated according to the laws of ordinary algebra with one exception. Addition is to be done modulo two:

$$1\,X^a + 1\,X^a = 0\,X^a \qquad 1\,X^a + 0\,X^a = 1\,X^a = 0\,X^a + 1X^a$$
$$0\,X^a + 0\,X^a = 0\,X^a \qquad\qquad -1\,X^a = 1\,X^a.$$

For example:

$$\begin{array}{rl}
\textit{addition} & \textit{multiplication} \\
1 + X \phantom{{}+X^2} + X^3 + X^4 & 1 + X \phantom{{}+X^2} + X^3 + X^4 \\
X + X^2 \phantom{{}+X^3} + X^4 & 1 + X \\ \cline{2-2}
1 + X + X^2 \phantom{{}+X^3} + X^4 & 1 + X \phantom{{}+X^2} + X^3 + X^4 \\ \cline{1-1}
X \phantom{{}+X^2} + X^3 + X^4 & X + X^2 \phantom{{}+X^3} + X^4 + X^5 \\ \cline{2-2}
 & 1 \phantom{{}+X} + X^2 + X^3 \phantom{{}+X^4} + X^5
\end{array}$$

In addition to the associative, distributive, and commutative properties of polynomials under this kind of algebra, we have, as in ordinary algebra, unique factorization; that is, every polynomial can be factored into prime or irreducible factors in only one way.[13]

Algebraic Description of Cyclic Codes

A cyclic code is defined in terms of a generator polynomial $P(X)$ of degree $n-k$. A polynomial of degree less than n is a code polynomial, *i.e.*, acceptable for transmission, if and only if it is divisible by the generator polynomial $P(X)$.[14] With this definition, the sum of two code polynomials is also a code polynomial, for if $F_1(X)$ and $F_2(X)$ are polynomials of degree less than n, which are divisible by $P(X)$, then $F_1(X)+F_2(X)$ is also of degree less than n and divisible by $P(X)$. Therefore, these codes are a special case of group codes, as studied by Slepian.[15]

If $P(X)$ has X as a factor, then every code polynomial has X as a factor and, therefore, has its zero-order coefficient equal to zero. Since such a symbol would be useless, we will consider only codes for which $P(X)$ is not divisible by X.

Code polynomials can be formed by simply multiplying any polynomial of degree less than k by $P(X)$. The following method has the advantage, however, that it results in a code polynomial in which the high-order coefficients are message symbols and the low-order coefficients are check symbols. To encode a message polynomial $G(X)$, we divide $X^{n-k}G(X)$ by $P(X)$ and then add the remainder $R(X)$ resulting from this division to $X^{n-k}G(X)$ to form the code polynomial:

$$X^{n-k}G(X) = Q(X)P(X) + R(X),$$

where $Q(X)$ is the quotient and $R(X)$ the remainder resulting from dividing $X^{n-k}G(X)$ by $P(X)$. Since in modulo two arithmetic, addition and subtraction are the same,

$$\boxed{F(X) = X^{n-k}G(X) + R(X) = Q(X)P(X),}$$

which is a multiple of $P(X)$ and, therefore, a code polynomial. Furthermore, $R(X)$ has degree less than $n-k$, and $X^{n-k}G(X)$ has zero coefficients in the $n-k$ low-order terms. Thus the k highest-order coefficients of $F(X)$ are the same as the coefficients of $G(X)$, which are the message symbols. The low order $n-k$ coefficients of $F(X)$ are the coefficients of $R(X)$, and these are the check symbols.

Example: Consider a code for which $n=15$, $k=10$, and $n-k=5$ which uses the generator polynomial $P(X)=1+X^2+X^4+X^5$. To encode the message 1010010001 corresponding to the polynomial $G(X)=1+X^2+X^5+X^9$, we divide $X^5G(X)$ by $P(X)$ and find the remainder. By long division it can be found that

$$X^5 + X^7 + X^{10} + X^{14} = (1 + X^2 + X^4 + X^5)$$
$$\cdot(1 + X + X^2 + X^3 + X^7 + X^8 + X^9) + (1 + X).$$

The code polynomial is formed by adding the remainder $(1+X)$ to $X^5G(X)$:

[11] J. E. Meggitt, "Error correcting codes for correcting bursts of errors," *IBM J. Res. Dev.*, vol. 4, pp. 329–334; July, 1960.

[12] W. W. Peterson, "Error Correcting and Error Detecting Codes," Technology Press, Cambridge, Mass., to be published.

[13] See, for example, R. D. Carmichael, "Introduction to the Theory of Groups of Finite Order," Dover Publications, Inc., New York, N. Y., p. 256; 1956.

[14] According to the usual definition, a cyclic code is a group code with the added property that the cyclic shift of a code vector is also a code vector. Codes obtained by making a number of the leading information symbols identically zero and dropping them are called shortened cyclic codes. The codes described in this paper are cyclic codes if X^m-1 is evenly divisible by $P(X)$, and otherwise are shortened cyclic codes. See Prange, footnote 1, and Peterson, footnote 12.

[15] D. Slepian, "A class of binary signaling alphabets," *Bell Sys. Tech. J.*, vol. 35, pp. 203–234; January, 1956.

$$F(X) = (1 + X) + (X^5 + X^7 + X^{10} + X^{14})$$

$$\underbrace{1\ 1\ 0\ 0\ 0}_{\text{check symbols}}\ \underbrace{1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1}_{\text{information symbols}}.$$

Principles of Error Detection and Error Correction

An encoded message containing errors can be represented by

$$H(X) = F(X) + E(X)$$

where $F(X)$ is the correct encoded message and $E(X)$ is a polynomial which has a nonzero term in each erroneous position. Because the addition is modulo two, $F(X)+E(X)$ is the true encoded message with the erroneous positions changed.

If the received message $H(X)$ is not divisible by $P(X)$, then clearly an error has occurred. If, on the other hand, $H(X)$ is divisible by $P(X)$, then $H(X)$ is a code polynomial and we must accept it as the one which was transmitted, even though errors may have occurred. Since $F(X)$ was constructed so that it is divisible by $P(X)$, $H(X)$ is divisible by $P(X)$ if and only if $E(X)$ is also. Therefore, an error pattern $E(X)$ is detectable if and only if it is not evenly divisible by $P(X)$. To insure an effective check, the generator polynomial $P(X)$ must be chosen so that no error pattern $E(X)$ which we wish to detect is divisible by $P(X)$.

$\neq 0$ erreur

To detect errors, we divide the received, possibly erroneous, message $H(X)$ by $P(X)$ and test the remainder. If the remainder is nonzero, an error has been detected. If the remainder is zero, either no error or an undetectable error has occurred.

$= 0$ erreur 0 erreur ou indétec. errn.

Example:

$$\begin{aligned} F(X) &= 1 + X + X^5 + X^7 + X^{10} + X^{14} \\ &= 1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1, \\ E(X) &= X^3 + X^6 + X^7 \\ &= 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0, \\ H(X) &= F(X) + E(X) \\ &= 1 + X + X^3 + X^5 + X^6 + X^{10} + X^{14} \\ &= 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1. \end{aligned}$$

This $F(X)$ was taken from the previous example. The remainder after $H(X)$ is divided by $P(X)=1+X^2+X^4+X^5$ is $X^2+X^3+X^4$, and the fact that this is not zero shows that an error must have occurred. The same remainder occurs if $E(X)$ is divided by $P(X)$, since $F(X)$ is divisible by $P(X)$.

The ability of a code to correct errors is related to its ability to detect errors. For example, any code which detects all double errors is capable of correcting any single error. This can be seen by noting that if only a single error occurs, we can try to correct it by trying to change each symbol. A polynomial with one error and one symbol changed can be a code polynomial only if the erroneous symbol is the one which was changed, since all other combinations are equivalent to double errors and, therefore, are detectable. Similarly, a code which detects all combinations of $2t$ errors can correct any combination of t errors, since if t or fewer errors occur, changing all combinations of t or fewer positions results in a code polynomial only if all the erroneous positions are changed. The same argument shows that any code capable of detecting any two error bursts of length b or less can correct any single burst of length b or less. Finally, the converse of these statements is also true; any t-error correcting code can detect any combination of $2t$ errors and any code capable of correcting any single burst of length b can be used instead to detect any combination of two bursts of length b.

Detection of Single Errors

Theorem 1: A cyclic code generated by any polynomial $P(X)$ with more than one term detects all single errors.

Proof: A single error in the ith position of an encoded message (counting from the left and numbering the leftmost position zero) corresponds to an error polynomial X^i. To assure detection of single errors, it is necessary only to require that $P(X)$ does not divide X^i evenly. Certainly no polynomial with more than one term divides X^i evenly. Q.E.D.

The simplest polynomial with more than one term is $1+X$:

Theorem 2: Every polynomial divisible by $1+X$ has an even number of terms.

Proof: Let $F(X)=X^a+X^b+X^c+ \cdots =(1+X)Q(X)$. Substituting $X=1$ gives

$$F(1) = 1 + 1 + 1 + \cdots = (1 + 1)Q(1) = 0.$$

There is one "1" in $F(1)$ for each term, and since the sum is zero, there must be an even number of terms. Q.E.D.

It follows that the code generated by $P(X)=1+X$ detects not only any single error, but also any odd number of errors. In fact, the check symbol must simply be an over-all parity check, chosen to make the number of ones in the code polynomial even.

Any polynomial of the form $1+X^c$ contains a factor $1+X$ since $1+X^c=(1+X)(X^{c-1}+X^{c-2}+ \cdots +1)$. Therefore, if $P(X)$ contains a factor $1+X^c$, any odd number of errors will be detected.

Double and Triple Error Detecting Codes (Hamming Codes)

A polynomial $P(X)$ is said to belong to an exponent e if e is the least positive integer such that $P(X)$ evenly divides $X^e-1(=X^e+1 \bmod 2)$.

Theorem 3: A code generated by the polynomial $P(X)$ detects all single and double errors if the length n of the code is no greater than the exponent e to which $P(X)$ belongs.

Proof: Detection of all double errors requires that

$P(X)$ does not evenly divide X^i+X^j for any $i, j<n$. We can factor X^i+X^j (assuming $i<j$) to $X^i(1+X^{j-i})$. It is sufficient to require that $P(X)$ should not divide $1+X^{j-i}$, since $P(X)$ is assumed not to be divisible by X. But $j-i<n\leqq e$, and therefore, since $P(X)$ belongs to the exponent e, $P(X)$ cannot divide $1+X^{j-i}$. Thus the code will detect double errors. Since $P(X)$ is not divisible by X and certainly could not be just the constant 1, it must have more than one term, and will, by Theorem 1, detect single errors also. Q.E.D.

It can be shown that for any m there exists at least one polynomial $P(X)$ of degree m that belongs to $e=2^m-1$. This is the maximum possible value of e. Polynomials with this property (usually called primitive polynomials) are always irreducible. A few such polynomials are listed in Appendix II, and more extensive tables are available.[12,16] Thus for any m there is a double-error detecting code of length $n=2^m-1$ generated by a polynomial $P(X)$ of degree m, which therefore, has m check symbols and 2^m-1-m information symbols. These codes can be shown to be completely equivalent to Hamming single-error correcting codes.[2,3,12,17]

Theorem 4: A code generated by $P(X)=(1+X)P_1(X)$ detects all single, double, and triple errors if the length n of the code is no greater than the exponent e to which $P_1(X)$ belongs.

Proof: The single and triple errors are detected by the presence of the factor $1+X$, as is shown by Theorem 2, and double errors are detected because $P_1(X)$ belongs to the exponent $e\geqq n$, exactly as in Theorem 3. Q.E.D.

Codes of maximum length result if $P_1(X)$ is a primitive polynomial, and these codes are equivalent to Hamming single-error correcting, double-error detecting codes.[1,2,15]

Detection of a Burst-Error

A burst-error of length b will be defined as any pattern of errors for which the number of symbols between the first and last errors, including these errors, is b.

Example:

$$\text{The } E(X) = X^3 + X^6 + X^7$$

$$= 0\,0\,0\,\underbrace{1\,0\,0\,1\,1}\,0\,0\,0\,0\,0\,0$$

of the previous example is a burst of length 5.

Theorem 5: Any cyclic code generated by a polynomial of degree $n-k$ detects any burst-error of length $n-k$ or less.

Proof: Clearly, any burst-error polynomial can be factored into the form $E(X)=X^iE_1(X)$ where $E_1(X)$ is of degree $b-1$. This burst can be detected if $P(X)$ does not evenly divide $E(X)$. Since $P(X)$ is assumed not to have X as a factor, it could divide $E(X)$ only if it could divide $E_1(X)$. But if $b\leqq n-k$, $P(X)$ is of higher degree than $E_1(X)$ and, therefore, certainly could not divide $E_1(X)$. Q.E.D.

A high percentage of longer bursts are detected as well.

Theorem 6: The fraction of bursts of length $b>n-k$ that are undetected is

$$2^{-(n-k)} \text{ if } b > n-k+1, \quad 2^{-(n-k-1)} \text{ if } b = n-k+1.$$

Proof: The error pattern is $E(X)=X^iE_1(X)$ where $E_1(X)$ has degree $b-1$. Since $E_1(X)$ has terms X^0 and X^{b-1}, there are $b-2$ terms X^j, where $0<j<b-1$, that can have either zero or one coefficients, and so there are 2^{b-2} distinct polynomials $E_1(X)$.

The error is undetected if and only if $E_1(X)$ has $P(X)$ as a factor.

$$E_1(X) = P(X)Q(X).$$

Since $P(X)$ has degree $n-k$, $Q(X)$ must have degree $b-1-(n-k)$. If $b-1=n-k$, then $Q(X)=1$, and there is only one $E_1(X)$ which results in one undetected error, namely $E_1(X)=P(X)$. The ratio of the number of undetected bursts to the total number of bursts is, therefore, $1/2^{b-2}=2^{-(n-k-1)}$ for this case. If $b-1>n-k$, $Q(X)$ has terms X^0 and $X^{b-1-(n-k)}$ and has $b-2-(n-k)$ arbitrary coefficients. There are, therefore, $2^{b-2-(n-k)}$ choices of $Q(X)$ which give undetectable error patterns. The ratio for this case is $2^{b-2-(n-k)}/2^{b-2}=2^{-(n-k)}$. Q.E.D.

Detection of Two Bursts of Errors (Abramson and Fire Codes)

Theorem 7: The cyclic code generated by $P(X)=(1+X)P_1(X)$ detects any combination of two burst-errors of length two or less if the length of the code, n, is no greater than e, the exponent to which $P_1(X)$ belongs.

Proof: There are four types of error patterns.

1) $E(X) = X^i + X^j$

2) $E(X) = (X^i + X^{i+1}) + X^j$

3) $E(X) = X^i + (X^j + X^{j+1})$

4) $E(X) = (X^i + X^{i+1}) + (X^j + X^{j+1})$

2) and 3) have odd numbers of errors and so they are detected by the $1+X$ factor in $P(X)$. For 4), $E(X)=(1+X)(X^i+X^j)$. The $1+X$ factor is cancelled by the $1+X$ factor in $P(X)$ so we will require for both 1) and 4) that X^i+X^j is not evenly divisible by $P_1(X)$. X^i+X^j is not evenly divisible by $P_1(X)$ as is shown in the proof of Theorem 3. Q.E.D.

These codes are equivalent to the Abramson codes, which correct single and double adjacent errors.[2,3] They are also the same as the Hamming single-error correcting, double-error detecting codes of Theorem 6.

Theorem 8: The cyclic code generated by

$$P(X) = (X^c + 1)P_1(X)$$

will detect any combination of two bursts

[16] A. A. Albert, "Fundamental Concepts of Higher Algebra," University of Chicago Press, Chicago, Ill.; 1956. This book contains a table of irreducible polynomials giving the exponent e to which they belong (see p. 161).

[17] N. M. Abramson, "A note on single error correcting binary codes," IRE Trans. on Information Theory, vol. IT-6, pp. 502–503; September, 1960.

$$E(X) = X^iE_1(X) + X^jE_2(X),$$

provided $c+1$ is equal to or greater than the sum of the lengths of the bursts, $P_1(X)$ is irreducible and of degree at least as great as the length of the shorter burst, and provided the length of the code is no greater than the least common multiple of c and the exponent e to which $P_1(X)$ belongs.

The proof, which is elementary but rather long, is given in Appendix IV. These are Fire codes.[4,12]

Other Cyclic Codes

There are several important cyclic codes which have not been discussed. Burst-error correcting codes have been treated also by Melas,[5] Meggitt,[11] and Reiger.[6] Codes for correcting independent random errors have been discovered by Melas,[10] Prange,[1] and Bose and Chaudhuri.[9,12,13] Any of these codes can also be used for error detection. The Bose-Chaudhuri codes are particularly important. For any choice of m and t there exists a Bose-Chaudhuri code of length 2^m-1 which is capable of correcting any combination of t errors (or alternatively, detecting any combination of $2t$ errors) and which requires a generator polynomial of degree no greater than mt. The description of the structure of these codes and the methods for choosing the polynomials is beyond the scope of this paper.

Implementation

Thus far, an algebraic method has been given for encoding and decoding to detect various types of errors. Briefly, to encode a message, $G(X)$, $n-k$ zeros are annexed (*i.e.*, the multiplication $X^{n-k}G(X)$ is performed) and then $X^{n-k}G(X)$ is divided by a polynomial $P(X)$ of degree $n-k$. The remainder is then subtracted from $X^{n-k}G(X)$. (It replaces the $n-k$ zeroes.) This encoded message is divisible by $P(X)$, but we have shown that if $P(X)$ is properly chosen, the message will not be evenly divisible if it contains detectable errors. The only nontrivial manipulation to be performed for both encoding and error detection is division by a fixed polynomial, $P(X)$.

The following is an example of division under addition modulo two:

$$\begin{array}{r}
1\,X^3+1\,X^2+0\,X+1 \\
1\,X^2+0\,X+1\,\overline{\smash{)}\,1\,X^5+1\,X^4+1\,X^3+0\,X^2+1\,X+0} \\
\underline{1\,X^5+0\,X^4+1\,X^3} \\
1\,X^4+0\,X^3+0\,X^2+1\,X+0 \\
\underline{1\,X^4+0\,X^3+1\,X^2} \\
0\,X^3+1\,X^2+1\,X+0 \\
\underline{1\,X^2+0\,X+1} \\
1\,X+1
\end{array}$$

We now repeat this division employing only the coefficients of the polynomials:

$$\begin{array}{r}
1\;1\;0\;1 \\
1\;0\;1\,\overline{\smash{)}\,1\;1\;1\;0\;1\;0} \\
\underline{1\;0\;1} \\
1\;0\;0\;1\;0 \\
\underline{1\;0\;1} \\
0\;1\;1\;0 \\
\underline{1\;0\;1} \\
1\;1
\end{array}$$

It can be seen that modulo two arithmetic has simplified the division considerably. Furthermore, we do not require the quotient, so the division to find the remainder can be described as follows:

1) Align the coefficient of the highest degree term of the divisor and the coefficient of the highest degree term of the dividend and subtract (the same as addition).
2) Align the coefficient of the highest degree term of the divisor and the coefficient of the highest degree term of the difference and subtract again.
3) Repeat the process until the difference has lower degree than the divisor. The difference is the remainder.

The hardware to implement this algorithm is a shift register and a collection of modulo two adders. (A modulo two adder is equivalent to the logical operation EXCLUSIVE OR). The number of shift register positions is equal to the degree of the divisor, $P(X)$, and the dividend is shifted through high order first and left to right. As the first one (the coefficient of the high-order term of the dividend) shifts off the end we subtract the divisor by the following procedure:

1) In the subtraction the high-order terms of the divisor and the dividend always cancel. As the high-order term of the dividend is shifted off the end of the register, this part of the subtraction is done automatically.
2) Modulo two adders are placed so that when a *one* shifts off the end of the register, the divisor (except the high-order term which has been taken care of) is subtracted from the contents of the register. The register then contains a difference that is shifted until another *one* comes off the end and then the process is repeated. This continues until the entire dividend is shifted into the register.

Fig. 1 gives a register that performs a division by $1+X^2+X^4+X^5$. Note that if alignment of divisor and dividend is considered to be accomplished when the high-order term of the dividend shifts off the end, then the divisor is automatically subtracted.

[13] W. W. Peterson, "Encoding and error-correction procedures for the Bose-Chaudhuri codes," IRE Trans. on Information Theory, vol. IT-6, pp. 459–470; September, 1960.

The shift register shown in Fig. 1 has, if used for encoding, one drawback that can be overcome by a slight modification. Recall that when encoding a message polynomial, $G(X)$, we calculate the remainder of the division of $X^{n-k}G(X)$ by $P(X)$. The straightforward procedure is to shift the message followed by $n-k$ zeroes into the register. When the last *zero* is in the register we obtain the remainder. Because this remainder replaces the $n-k$ zeros to form the encoded message, it is necessary to delay the message $n-k$ shift times so that the remainder can be gated in from the encoder register at the proper time.

An example of this method of encoding is given in Fig. 2. Initially, the gate G_1 is open and the gate G_2 is shorted, allowing the remainder on dividing $X^{n-k}G(X)$ to be calculated. After the message plus $n-k$ *zeros* is shifted in, G_1 is shorted and G_2 is opened. This allows the remainder which is now in the register to replace the $n-k$ zeros in the output. Error detection with this circuit requires that gate G_1 be open and gate G_2 be shorted. After $H(X)$ has been shifted in, the register contains the remainder. If this is nonzero, an error has occurred.

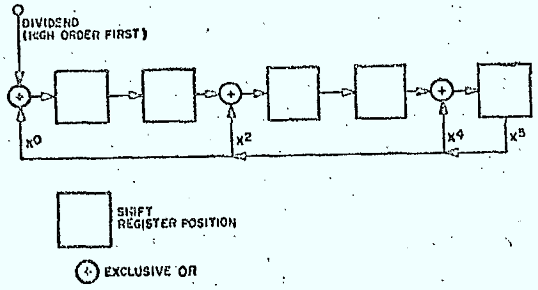


Fig. 1—A shift register for dividing by $1+X^2+X^4+X^5$.

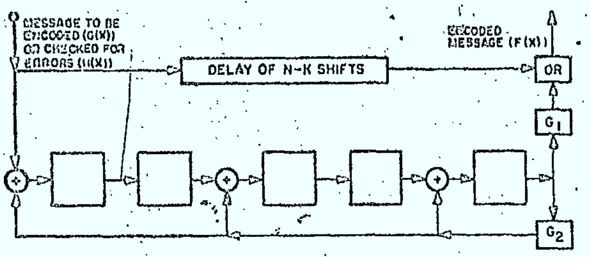


Fig. 2—One method of encoding on detecting errors. (In this example, $P(X)=1+X^2+X^4+X^5$.)

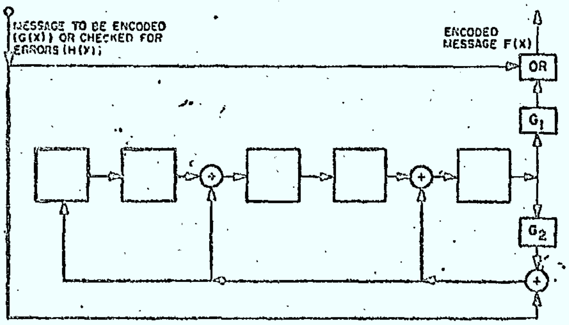


Fig. 3—A more efficient circuit for encoding and error detection. (In this example, $P(X)=1+X^2+X^4+X^5$.)

The delay of $n-k$ shifts can be avoided if Fig. 2 is modified to give the circuit of Fig. 3. In Fig. 3, instead of shifting the polynomial into the low-order end of the register, it is treated as if it were shifting out of the high-order end. This is equivalent to advancing every term in the polynomial by $n-k$ positions, or multiplying by X^{n-k}. Now in encoding, as soon as $G(X)$ has been completely shifted into the register, the register contains the remainder on dividing $X^{n-k}G(X)$ by $P(X)$. Then gate G_1 is shorted, gate G_2 is opened, and the remainder follows the undelayed $G(X)$ out of the encoder to form $F(X)$.

To minimize hardware, it is desirable to use the same register for both encoding and error detection, but if the circuit of Fig. 3 is used for error detection we will get the remainder on dividing $X^{n-k}H(X)$ by $P(X)$ instead of the remainder on dividing $H(X)$ by $P(X)$. It turns out that this makes no difference, for if $H(X)$ is evenly divisible by $P(X)$ then obviously $H(X)X^{n-k}$ is evenly divisible, and if $H(X)$ is not evenly divisible by $P(X)$ then $H(X)X^{n-k}$ will not be evenly divisible either, provided the divisor $P(X)$ does not have a factor X. Any useful $P(X)$ will satisfy this restriction. The circuit of Fig. 3 can, then, be used for both encoding and error detection.

Error correction is by its nature a much more difficult task than error detection. It can be shown that each different correctable error pattern must give a different remainder after division by $P(X)$. Therefore, error correction can be done as follows:

1) Divide the received message $H(X)=F(X)+E(X)$ by $P(X)$ to obtain the remainder.
2) Obtain the $E(X)$ corresponding to the remainder from a table or by some calculation.
3) Subtract $E(X)$ from $H(X)$ to obtain the correct transmitted message $F(X)$.

Both the encoding and step 1 of the decoding are the same for error correction as for error detection. The error-correction equipment is more complex in that it requires equipment for the table look-up or computation of step 2, and it requires that the entire received message $H(X)$ be stored temporarily while the remainder is being calculated and $E(X)$ is being determined. The calculation required in step 2 can be done simply with a shift register for burst-error or single-error correcting codes, but is quite complex for codes that correct multiple random errors. Details of error-correction procedures are beyond the scope of this paper, but can be found in references.[11,12,17]

Conclusion

A simple presentation of cyclic codes has been given in terms of polynomials. The attractive features of these codes for error detection, both their high efficiency and the ease of implementation, have been emphasized.

Appendix I

Notation

k = number of binary digits in the message before encoding,

n = number of binary digits in the encoded message,

$n-k$ = number of check digits,

b = length of a burst of errors,

$G(X)$ = message polynomial (of degree $k-1$),

$P(X)$ = generator polynomial (of degree $n-k$),

$R(X)$ = remainder on dividing $X^{n-k}G(X)$ by $P(X)$, $R(X)$ is of degree less than $n-k$,

$F(X)$ = encoded message polynomial, $F(X) = X^{n-k}G(X) - R(X)$,

$E(X)$ = error polynomial,

$H(X)$ = received encoded message polynomial, $H(X) = F(X) + E(X)$.

Appendix II

A Short Table of Primitive Polynomials

Primitive Polynomial	e
$1+X$	1
$1+X+X^2$	3
$1+X+X^3$	7
$1+X+X^4$	15
$1+X^2+X^5$	31
$1+X+X^6$	63
$1+X^3+X^7$	127
$1+X^2+X^3+X^4+X^8$	255
$1+X^4+X^9$	511
$1+X^3+X^{10}$	1023
$1+X^2+X^{11}$	2047
$1+X+X^4+X^6+X^{12}$	4095
$1+X+X^3+X^4+X^{13}$	8191
$1+X+X^6+X^{10}+X^{14}$	16383
$1+X^{14}+X^{15}$	32767

Appendix III

Data for Some Representative Codes

Detection Capabilities	$k_{\max}$	$n-k$	$P(X)$	Reference
Any odd number of errors	any value	1	$1+X$	Theorem 2
Two errors, a burst of length 4 or less, 88 per cent of the bursts of length 5, 94 per cent of longer bursts*	11	4	$1+X+X^4$	Theorems 3, 5, 6
Two errors, a burst of 9 or less, 99.6 per cent of the bursts of length 10, 99.8 per cent of longer bursts	502	9	$1+X^4+X^9$	Theorems 3, 5, 6
Two bursts of length 2 or less, any odd number of errors, a burst of 5 or less, 93.8 per cent of the bursts of length 6, 96.9 per cent of longer bursts†	10	5	$(1+X+X^4)(1+X)=1+X^2+X^4+X^5$	Theorems 2, 5, 6, 7
Two bursts of combined length 12 or less, any odd number of errors, a burst of 22 or less, 99.99996 per cent of the bursts of length 23, 99.99998 per cent of longer bursts	22495	22	$(1+X^2+X^{11})(1+X^{11})=1+X^2+X^{13}+X^{22}$	Theorems 2, 5, 6, 8
Any combination of 6 or fewer errors, a burst of length 11 or less, 99.9 per cent of bursts of length 12, 99.95 per cent of longer bursts	12	11	$1+X^2+X^4+X^5+X^6+X^{10}+X^{11}$	Theorems 5, 6, and footnote 1
Any combination of 7 or fewer errors, any odd number of errors, a burst of length 31 or less, all but about 1 in 10^9 of longer bursts	992	31	$(1+X)(1+X^3+X^{10})$ $(1+X+X^2+X^3+X^{10})$ $(1+X^2+X^3+X^8+X^{10})$	Theorems 2, 5, 6, and footnotes 9, 12, 18

° *Note:* $1+X+X^4$ belongs to $e=15$ and $11+4=15$.

† *Note:* This is the code used in all examples.

Appendix IV

Proof of Theorem 8

The error polynomial has the form:

$$E(X) = X^i[E_1(X) + X^{j-i}E_2(X)].$$

$E_1(X)$ has degree b_1-1 and $E_2(X)$ has degree b_2-1.

The generator polynomial $P(X)$ cannot have a factor X so we need only consider the factor of $E(X)$ in brackets. Let $j-i=d$, assume $E'(X)=E_1(X)+X^dE_2(X)$ is divisible by X^c+1, and let $d=cq+r$ with $r<c$.

Then,

$$\begin{aligned} E'(X) &= E_1(X) + X^{cq+r}E_2(X) \\ &= E_1(X) + X^rE_2(X) + [X^rE_2(X)]\cdot[X^{cq}+1]. \quad (1) \end{aligned}$$

Now $X^{cq}+1$ contains a factor X^c+1 for

$$\begin{aligned} X^{cq}+1 = (X^c+1)(X^{c(q-1)} &+ X^{c(q-2)} \\ &+ X^{c(q-3)} + \cdots + X^0). \end{aligned}$$

Hence the rightmost term in (1) is divisible by X^c+1. $E'(X)$ was assumed divisible by X^c+1 and so from (1), $E_1(X)+X^rE_2(X)$ must be divisible by X^c+1. Using this result, we can let

$$E_1(X) + X^rE_2(X) = [X^c+1][Q(X)]$$

$$E_1(X) + X^rE_2(X) = Q(X) + X^cQ(X). \qquad (2)$$

We will assume that $Q(X)\neq 0$. Let the degree of $Q(X)$ be h. The degree of the right-hand side of (2) is $c+h$ and the degree of the left-hand side is either b_1-1 or $r+b_2-1$. Then, for (2) to be true we must have either $c+h=b_1-1$ or $c+h=r+b_2-1$. Since it was assumed that $c\geqq b_1+b_2-1$ we must have the second relation.

$$c + h = r + b_2 - 1.$$

Again using $c\geqq b_1+b_2-1$ we have

$$b_1 + b_2 - 1 + h \leqq r + b_2 - 1 \quad \text{or} \quad b_1 + h \leqq r.$$

From this, $b_1\leqq r$ or $b_1-1<r$ and as $b_1\neq 0$, $h<r$.

Applying these results to (2), we see that both $E_1(X)$ and $Q(X)$ are of lower degree than any of the terms in $X^rE_2(X)$. It follows then, given the assumption that $Q(X)\neq 0$, that

$$X^rE_2(X) = X^cQ(X). \qquad (3)$$

As $E_2(X)$ always contains an X^0 term, the lowest order term in $X^rE_2(X)$ is of degree r. The lowest order term in $X^cQ(X)$ is of degree at least c but $r<c$ so (3) can never be satisfied. Therefore, the only solution of (2) is with $Q(X)=0$ giving $E_1(X)+X^rE_2(X)=0$.

As $E_1(X)$ always contains an X^0 term, $r=0$ and $E_1(X)=E_2(X)$. Substituting in (1) gives

$$E'(X)=E_2(X)[X^{cq}+1].$$

This is the form of the error polynomial if it is evenly divisible by X^c+1. It is sufficient to show that this polynomial is not evenly divisible by $P_1(X)$ to guarantee that $E(X)$ is never evenly divisible by $P(X)=P_1(X)$, $[X^c+1]$. $P_1(X)$ is irreducible, so to divide $E'(X)=E_2(X)$ $[X^{cq}+1]$ it must divide one of the factors. For this special case, $E_1(X)=E_2(X)$ so $b_1=b_2$, and since both bursts have the same length, this is the length of the shorter burst. It was specified that $P_1(X)$ is of degree no less than the length of the shorter burst so it is of higher degree than $E_2(X)$ and cannot divide $E_2(X)$.

It remains to show that $P_1(X)$ does not evenly divide $X^{cq}+1$. Make the substitution $cq=ue+v$ where e is the exponent to which $P_1(X)$ belongs and $v<e$. Now $v\neq 0$ because cq is less than or equal to the length of the message and the length of the message is less than or equal to the least common multiple of c and e. Since cq is a multiple of c, it cannot be a multiple of e.

$$X^{cq} + 1 = X^{ue+v} + 1$$

$$X^{cq} + 1 = X^v + 1 + X^v(X^{ue} + 1).$$

As was shown previously, $X^{ue}+1$ is divisible by X^e+1. Furthermore, $P_1(X)$, by definition, divides X^e+1; therefore, $P_1(X)$ divides $X^{ue}+1$. However, X^e+1 is the lowest degree polynomial of this form that $P_1(X)$ divides, so $P_1(X)$ does not divide X^v+1. As $v\neq 0$, we have shown that $P_1(X)$ does not divide $X^{cq}+1$, completing the proof.

General References

TEXTBOOK REFERENCES

1. R. Watson — Time Sharing System Design Concepts
McGraw Hill 1970 ($12.50)

2. Harry Katzan Jr. — Advanced Programming
Van Nostrand Reinhold 1970 ($15.00)

3. J. Martin — Teleprocessing Network Organization
Prentice Hall 1970

4. DATAMATION — A Catalogue of EDP Products of Services
(1971)
Available from Datamation, 1301 South Grove Ave.,
Barrington, Illinois 60010. ($35.00)

5. AUERBACH — On Time Sharing (1967)
available from Auerbach Info. Inc.,
Philadelphia, Pa 19109. ($14.00)

6. James Ziegler — Time Sharing Data Processing Systems
Prentice Hall 1967 ($13.00)

7. Douglas Parkhill — The Challenge of The Computer Utility.
Addison Wesley ($8.00)