TIME DIVISION MULTIPLEXING OF DIGITAL SIGNALS

by

WAYNE F. WESTGATE

B.S., Kansas State University, 1966

A THESIS

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Electrical Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1968

Approved by:

Floyd W. Harris

Major Professor

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

In the world today there is an ever growing need for communication; thus transmission systems are being built to try to fulfill the needs of our society. A great deal of communication tends to be in digital form, as digital information is more efficient to transmit. Therefore, time division multiplexing of digital information plays an important part in the communications world and will continue to become more important as the communication needs of society increase.

Fixed Frames and Variable Frames

A device which takes digital data from several sources, compresses the data in time if necessary, addresses and interleaves the data into a single high traffic stream, is termed a time division digital multiplexer. There are at least two ways in which digital multiplexing may be accomplished with efficiency. These may be distinguished from each other by the format used to scan the sources. The first might be called fixed frame length multiplexing, while the second could be called variable frame length multiplexing. In the multiplexing operation each input source is scanned and sampled according to the predetermined format. A set of samples from one scan of all sources is termed a frame. When a fixed frame is used time slots are reserved for each source on every scan. If the variable frame is employed the length of the frame may vary from scan to scan. This is the result of the fact that all sources may or may not have data to transmit. In the case of the fixed frame technique there is a one-to-one correspondence between the source and the time slot, while in the variable frame scheme there is no correspondence.

Probabilistic and Deterministic Sources

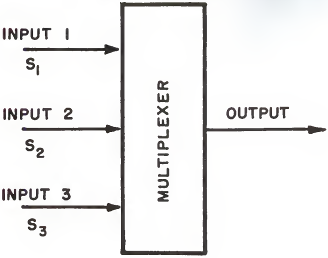
The type of multiplexing frame used is dependent upon the kind of data to be multiplexed. If the sources produce a continuous stream of digital information then the fixed frame type of multiplexing should be used. This choice of format is the most economical as far as electronic equipment and synchronizing information is concerned. If the sources of data give information at intermittent intervals, the variable frame scheme may be used. The advantage of the variable frame is that on a single scan time slots are reserved only for sources which have information to transmit. As the multiplexer scans each input, it may remain long enough on one source to send one entire word to the transmitter or it may sample the source only long enough to transmit one bit. In the case of the fixed frame, sampling is usually done on a bit basis, while for the variable frame, word or block sampling is usually used. The reason for choosing word sampling over bit sampling in the case of the variable frame scheme is that synchronization and addressing is more efficient when a word sampling scheme is used.

Synchronization

There are two kinds of synchronization which must be performed by a digital multiplexer. First, before multiplexing can take place, the incoming bit streams must be, or must be made to be, precise submultiples of some master frequency. There have been four proposed schemes to accomplish source synchronization for continuous data sources (10). These are the master clock, phase averaging, stable clocks, and pulse stuffing. Source synchronization of probabilistic data is accomplished by buffer store units placed in each input before the multiplexer. Second, to perform the demultiplexing operation additional information must be transmitted with the multiplexed stream. When using fixed frame multiplexing the only additional information needed is the synchronization pulse to keep the multiplexer and demultiplexer at the same frequency and in proper phase. Once the demultiplexer has locked on in proper phase with the bit stream, information as to which source the pulse came from is then given by the time slot position relative to the sync pulse. In variable frame multiplexing, an address code must be sent with each information block conveying to the demultiplexer the identification of the source from which the information block came. (See Figure I-1.)

The master clock approach to frequency locking sources which are at different geographical points involves the distribution of a synchronizing frequency to all interfaces (multiplex-demultiplex stations) from a central location. Thus, all sampling clocks and coders can be harmonically related to the "master frequency." There are several undesirable features about this system of synchronization for large numbers of interfaces. Because the master frequency is derived from only one centrally located position, it would have to be highly reliable with sufficient backup systems in case of failure. It would have to be protected against natural and manmade disasters, as interruption of the master clock would completely disable the whole system. Because of transmission line delay there would have to be variable delays built into the interfaces so that proper phasing could be accomplished between bit streams. The initial costs of the master clock system would be large if the number of interfaces was large and the various multiplexing-

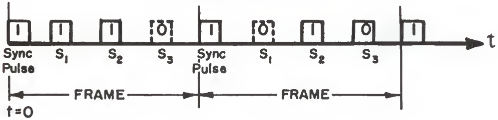
MULTIPLEXER WITH THREE INPUTS



ADDRESS CODE FOR VARIABLE LENGTH FRAME MULTIPLEXER

INPUT	ADDRESS
1	00
2	01
3	10

OUTPUT FOR FIXED FRAME LENGTH MULTIPLEXER WITH THREE INPUTS



OUTPUT FOR VARIABLE FRAME LENGTH MULTIPLEXER WITH THREE INPUTS

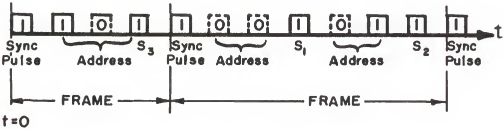


Figure I-1

demultiplexing stations were spread far apart. For a system which was very specialized and required few interfaces the "master clock" approach might well be feasible.

Phase averaging is the second technique for synchronizing the incoming pulse streams. This technique allows for sampling clocks of all coders to be frequency locked without the use of a master clock. At every interface there are pulse streams entering and leaving. A reference frequency may then be established from an average of all frequencies entering the multiplexer. A voltage controlled oscillator whose phase is locked to the average of all signals entering the various locations will then set a reference for all sampling frequencies throughout the network. It can be shown (10) that the reference frequency will be the same at all interfaces and that this frequency will be bounded by the highest and lowest free running oscillators in the network. This system is not as susceptible to failure as the master clock, but the initial expenses would still be high. The scheme has been analyzed in considerable detail by Bell Telephone Laboratories Systems Research Department, but before a complete system is committed further analysis should be carried out.

Stable clocks is the third method of synchronizing continuous incoming pulse streams. In this system a buffer storage is provided for each channel and a very stable oscillator is used as the sampling clock. Now if an input bit stream of frequency f_i is multiplexed to a frequency $f_i + \Delta f$, then the elastic store will be exhausted every $C/\Delta f$ seconds if there were C bits of storage. Defining the clock stability factor as $s = \Delta f/f_i$, then the store is exhausted every C/sf_i seconds. Each time the store is exhausted, information must either be lost or repeated in order for the store to recover.

As an example, consider a 100 Megabits/sec (Mb/s) pulse stream and a clock stability factor of 10^{-10}. If once a day reframing is allowed, 10^3 bits of storage will be needed. This would add great expense to the multiplexer. The stable clock system of synchronization although simple in concept is not usable if loss of information is not acceptable.

Pulse stuffing is the fourth method of synchronization. In the pulse stuffing technique, extra pulses which carry no information are inserted into the incoming bit streams to change the bit rates to some harmonic rate of the multiplexing frequency. The extra pulses are then removed at the demultiplexer. This technique requires the incoming bit streams to be nominally some harmonic of the multiplexing frequency. If the bit rates were not harmonics of the multiplexing frequency, excessive stuffing would be required to change the bit rates. The pulse stuffing technique is discussed in greater depth in Chapter II.

If the sources of information are probabilistic in nature, synchronization is accomplished by the use of buffer storage units. It is only necessary to be able to have an output bit rate which is equal to or greater than the sum of the long time average input bit rates. In order to make calculations and predictions about the probabilistic system it would be necessary to form a mathematical model of the system and then make predictions about the size of the storage and the number of sources which can be multiplexed.

Introduction to Chapters II, III, and IV

There are in general two systems of multiplexing which are under consideration in the present literature. The first system uses a fixed frame format, pulse stuffing for synchronization, and is called the asynchronous digital signal multiplexer by Johannes and McCullough (4). Chapter II is an analysis of the logic of the pulse stuffing technique and of the limitations of the system. The asynchronous system requires that all inputs be continuous and that all inputs be nominally submultiples of each other. The second system is called the digital data dynamic transmission system by Hasegawa, Tezuka, and Kasahara (3). The dynamic system entails the variable frame format and uses a phase averaging clock for synchronization. This system can be made to handle intermittent sources of diverse message frequencies or even continuous sources of different frequencies. The price paid for such a versatile system is transmission time; that is, an address must be sent with each sample.

Chapter III contains a description of the dynamic multiplexer, while Chapter IV presents a proposed mathematical model of the variable frame multiplexer. Each input to the multiplexer of Chapter IV is assumed to have Poisson distributed transitions with each input having a different statistical average rate. Next the total number of demands generated by all sources is assumed to be Poisson distributed with an average value of one-half the sum of the input statistical averages. This last assumption corresponds to over estimating the probabilistic rate at which demands occur. These ideas will be expanded upon in the presentation of the mathematical model. The main things which have been accomplished in this paper are a general organization of digital multiplexing and a presentation of a mathematical model of the variable frame system.

CHAPTER II

A CONTINUOUS FLOW, FIXED FRAME, DIGITAL TIME DIVISION MULTIPLEXER

The asynchronous multiplexer is a device which is able to combine several continuous bit streams into a single high speed stream. This multiplexer, however, requires that the nominal bit rates into the inputs be approximately integer multiples of each other. The reason for this is that extra pulses called "stuffed pulses" are inserted into the input bit streams to make them precise submultiples of each other. In this way, merging the input streams into a single stream is accomplished with ease. Because of the fixed frame and, more important because of pulse stuffing synchronization, the input data must be of a continuous nature, for if just one of the inputs should be interrupted, stuffed pulses would have to be inserted to equal the rate of the interrupted input. Since the maximum rate at which stuffing can take place is far less than any single input rate an interrupted input would completely disable the system. The maximum stuffing rate is a direct result of the method used to remove the stuffed pulses at the demultiplexer.

A simplified block diagram of the asynchronous digital signal multiplexer is shown in Figure II-1. Assume that the inputs to this multiplexer have bit rates f_1, f_2, f_3 . . ., f_h, f_{h+1}, f_{h+2} and that

$$n_1 f_1 = n_2 f_2 = n_3 f_3 = \ldots = n_h f_h = n_{h+1} f_{h+1} = n_{h+2} f_{h+2}$$

where n_1, n_2, . . . n_{h+2} are all integer numbers. The inputs f_{h+1} and f_{h+2}

A FIXED FRAME LENGTH MULTIPLEXER

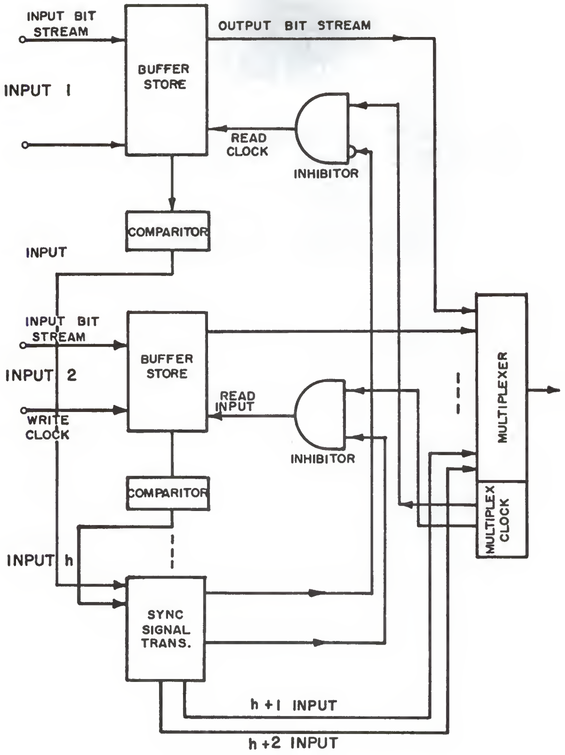


Figure II -1

are not external inputs but inputs through which synchronizing data and stuffed pulse data are sent. They may, however, be treated as external sources for purposes of analysis and will be discussed later in this chapter. Let f_1 be the highest bit rate and f_h the lowest. The multiplex clock in combination with the multiplexer scans and samples each buffer store at the nominal frequency f_o. One cycle of the scanner is governed by the format of a single frame. In a single frame there is one pulse from input h+1, one from h+2, one from input h, . . . n_h/n_2 from input 2, and n_h/n_1 from input 1. Because there cannot be such a thing as a fractional pulse, the ratios

$$\frac{n_h}{n_h}, \ . \ . \ . \ . \ , \frac{n_h}{n_3}, \frac{n_h}{n_2}, \frac{n_h}{n_1}$$

must also be integer numbers. The length of the frame is determined by the lowest frequency to be multiplexed, namely $T = 1/f_h$. In order to preserve continuity, nominally,

$$f_o = f_1 + f_2 + f_3 + \ . \ . \ . + f_h + f_{h+1} + f_{h+2}.$$

But it is possible that the output frequency, which is governed by the multiplex clock, may drift from its assigned frequency. It is also possible that the input frequencies f_1, f_2, . . . f_h may drift from their exact values. Therefore, to preserve continuity exactly

$$f_o \pm \delta_o = f_1 \pm \delta_1 + f_2 \pm \delta_2 + \ . \ . \ . + f_h \pm \delta_h + f_{h+1} + f_{h+2}.$$

To insure that there is no overflow the following inequality is made to hold

$$f_o - \delta_o > f_1 + \delta_1 + f_2 + \delta_2 + \ . \ . \ . + f_h + \delta_h + f_{h+1} + f_{h+2}. \qquad \text{(II-1)}$$

If this condition is valid, the storage units, into which the incoming bit streams are read, will on occasion be emptied because the output rate of information is greater than the input rates combined. If buffer h should be nearly depleted the comparator associated with buffer h signals the synchronizing transmitter which in turn inhibits the multiplexer clock from reading out the buffer store. At the same time, an extra pulse is stuffed in the time slot reserved in the frame for the h-th input. In this way, the buffer store may recover and bit integrity is preserved.

In order to keep the multiplexer and demultiplexer locked to the same frequency and in phase with each other, a synchronizing pulse is sent in the h+1 time slot in each frame. It has been suggested that an alternating bit pattern be used (4)(11). That is, in one frame a pulse is sent in the h+1 slot, then in the next frame no pulse is sent, and then in the next frame a pulse is again sent in the h+1 slot. Thus, an alternating pattern is set up which the demultiplexer electronics can easily recognize. The demultiplexer must also remove the stuffed pulses and, therefore, must receive information as to which channel was stuffed. The multiplexer sends stuffing information to the demultiplexer via the h+2 time slot. A typical frame sent by the multiplexer might look like that shown in Figure II-2. The format by which the stuffing information is sent is in itself lengthy and is called a stuffing frame. Since there is only one slot in each multiplex frame allocated to

TYPICAL FRAME SENT BY MULTIPLEXER

Figure II-2

sending stuffing information, it takes many multiplex frames to send one stuffing frame. Johannes and McCullough have suggested (4) that a second order M word be sent first. An n-th order M word for a binary channel is defined as a word capable of being detected and located exactly in time in the presence of n or less binary errors within a number of bits equal to the length of the M word. Following the M word is the stuffing information for the first four time slots of the multiplexer frame. The stuffing information as suggested by Witt (11) should be redundantly coded, such as "000" meaning no stuff and "111" meaning stuffed. The number of stuff words (called C words) depends on the number of slots in the multiplex frame. At the end of the stuffing frame is a word which marks the end and resets the logic for identifying the stuffing information for the first time slot. A more complete discussion of the stuffing format can be found in Witt (11) and Johannes and McCullough (4).

Let the stuffing frame be m bits long; then it would require m multiplex frames to send one stuffing frame to the demultiplexer. If the multiplex frame is T_f seconds long then the maximum stuffing rate per time slot in the multiplex frame is $1/mT_f$. Consider, for example, the input h, which has an input rate of $f_h \pm \delta_h$. Because the multiplexing frequency is set by the inequality of equation II-1, the multiplexer demands $f_h' \pm \delta_h'$ from input h. The stuff rate Δf_h for input h is given by

$$f_h' \pm \delta_h' - f_h \mp \delta_h = \Delta f_h$$

and if

$$f_h' \pm \delta_h' - f_h \mp \delta_h < 1/mT_f \qquad \text{(II-2)}$$

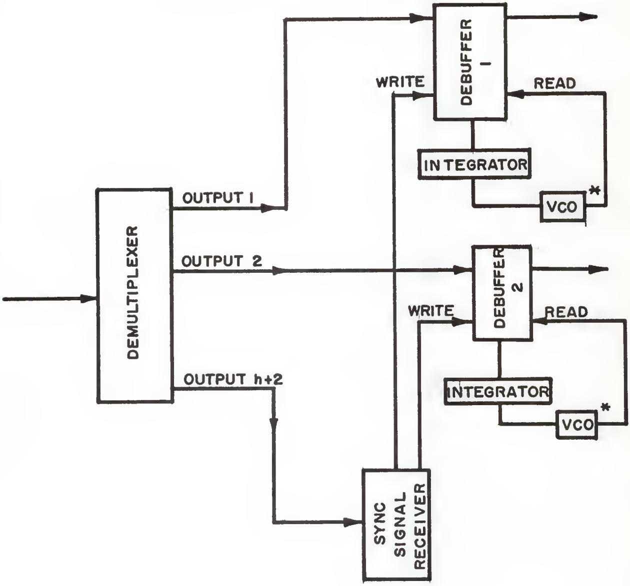
holds, then all stuffed pulses in time slot h will be removed at the demultiplexer provided that no other input exceeded its maximum stuffing rate. If the inequality does not hold, then some of the stuffed pulses will not be removed or a pulse may be removed that was in reality not stuffed. The maximum stuffing rate is one of the biggest limiting factors of the asynchronous digital multiplexer.

A block diagram of the demultiplexer is provided in Figure II-3. At the demultiplexer the sync signal receiver disassembles the multiplex frame and stores each bit in the appropriate buffer store. When a stuffed pulse is found in the frame, the sync signal receiver is inhibited so that the stuffed pulse is not read into the buffer store. In this way, the stuffed pulses are eliminated from the bit streams. The next step after the pulses are read into the memory is to read them out at their original frequency by a voltage controlled oscillator. A control voltage is derived from the buffer corresponding to the number of bits stored and is passed through an integrator to obtain the average number of bits stored which, in turn, controls the oscillator frequency. The information can then read out of the desynchronizing buffer at the same frequency that it was read into the synchronizing buffer. This method of reading the information out is also called a phase locked loop (11).

The apparent advantages of this system are that it requires little storage and, therefore, storage costs are not high. The multiplexer and demultiplexer are relatively independent of each other in that they do not have to have a frequency standard at each location but rather the sync signal receiver follows the sync signal transmitter. The whole system is at a disadvantage in that it can handle only those input rates which are integer multiples of each other and even more restrictive, the ratios

$$\frac{n_h}{n_h}, \ldots, \frac{n_h}{n_3}, \frac{n_h}{n_2}, \frac{n_h}{n_1}$$

THE DEMULTIPLEXER



* VCO — VOLTAGE CONTROLLED OSCILLATOR

Figure II-3

must be integer values in order to make a fixed frame possible. Another disadvantage is that the inputs must be continuous as interruption of one input interrupts the entire system. Therefore, a more versatile system is needed for sources of a wide range in frequencies and intermittent operation. This leads to the variable frame length multiplexer.

CHAPTER III

DESCRIPTION OF A VARIABLE FRAME DIGITAL TIME DIVISION MULTIPLEXING SCHEME

In the search for a more versatile system, one which can handle inputs which are intermittent and of a wide range of pulse rates, there have been several contributors. It appears that Filipowsky and Scherer (1) were the first to contrive the variable frame scheme. Ristenbatt and Rothschild (9) have worked with the problem of multiplexing of probabilistic digital data and have showed very promising results. There is also work being done by Najjar, Antonio and Blasbalg (6) on multiplexing inputs of diverse bit rates, which has not as yet been published. Other works by Hasegawa, Tezuka, and Kasahara (3) on the digital data dynamic transmission systems have shown good results. The subject of this chapter will be the dynamic time division multiplexer.

It would be possible for the dynamic multiplexer to handle continuous data of diverse input rates if the proper adjustments were made on the output rate. In the case of pulse stuffing synchronization, when a buffer became empty an extra pulse was stuffed into the waiting time slot so that continuity was preserved. In the dynamic multiplexer an empty buffer is simply skipped and information is sent from another buffer which has data to send. The price paid for such versatility is in addressing the data, for each block of information sent is addressed as to which input originated the block. It would seem that the dynamic time division multiplexer would be most applicable to a system in which the input data was probabilistic and the input rates were diverse.

A block diagram of the dynamic multiplexer is shown in Figure III-1. Let the inputs to the multiplexer be 1, 2, . . ., i, . . ., h where each input has a different bit rate f_1, f_2, . . ., f_i, . . . f_h. Let each input be random, that is, not all inputs are active at the same time. Suppose that the i-th input is active; then as information is received at the i-th input it is read into the i-th buffer at the rate of f_i bits/sec. The primary purposes of the buffer units are to synchronize the incoming data with data coming in on other inputs and to act as a ballast for overloads. As information flows into an empty buffer a signal C_i is put onto the "and gate" A_{1i} which indicates that buffer i has information in store to transmit. The output of the multiplexer clock is fed through an inhibitor, which is inactive at the present, to the ring counter. When the ring counter output reaches the "and gate" A_{1i}, the gate is opened and a signal C_i' activates the pulse generator 1 through the "or gate B_1." Both pulse generators 1 and 2 have as an output one pulse for each positive going signal at their inputs. The output pulse from pulse generator 1 sets the flip-flop through "or gate" B_4. The flip-flop activates the inhibitor which stops the clock pulse to the ring counter; therefore, the scan is stopped on A_{1i}. At the instant that C_i appears at the output of A_{1i}, it is also applied to "and gate" A_{2i} and if there is no information being transmitted at that instant then "and gate" A_{2i} is opened and an output appears from A_{2i} which sets the holding gate H_i. The output H_i does two things; first, it activates the binary invertor 1 through the "or gate" B_2 which in turn closes "and gate" A_{2i} and disables "and gates" A_{21}, A_{22}, . . . A_{2h}. In this way no other blocks of information can be read out until a block of information from buffer i has been transmitted. Second,

A VARIABLE LENGTH FRAME MULTIPLEXER

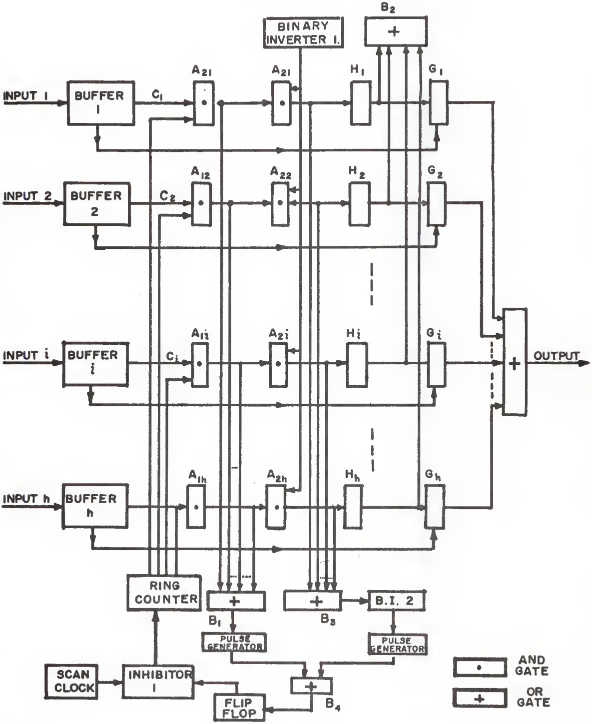


Figure III-1

the output from H_i opens gate G_i which allows a block of information to be read out, addressed and transmitted. When the binary invertor (B.I.) 1 is activated by the output from H_i and the gate A_{2i} is closed, then a positive going signal appears at the output of binary invertor 2. Thus, the pulse generator 2 is activated and the flip-flop is reset through B_4. The inhibitor is deactivated and the ring counter begins to scan again. As soon as the ring counter finds another buffer with information to transmit, it waits at that buffer until the previous block of information has been transmitted. In this way, the buffers are scanned at the same time that information is being transmitted and there is no lag between blocks of information in the output. The holding duration of the holding circuits is just long enough to allow one block of information and its address to be transmitted and then it automatically deactivates itself. This of course enables "and gates" A_{21}, A_{22}, . . . A_{2i}, . . . A_{2h} again, allowing the next block of information to start its holding time.

A block diagram of the logic at the demultiplexer is presented in Figure III-2. The address attached to the front of each block of information contains a start-stop signal. The start-stop signal excites the start-stop generator which sets the holding circuit H. The length of time that the holding circuit remains active is just long enough so that address may be read into the shift register. It can be seen that as long as the holding circuit H is active the inhibitor circuit 2 inhibits the address information from being passed on to the gates G_{21}, G_{22}, . . . G_{2i}, . . . G_{2h} and the "and gate" A_2 allows the write signal from the oscillator to pass into the shift register. The write oscillator's frequency is controlled by the frequency of the incoming data. As the address code is written into the shift register it

THE DEMULTIPLEXER FOR THE MULTIPLEXER OF FIGURE III-1

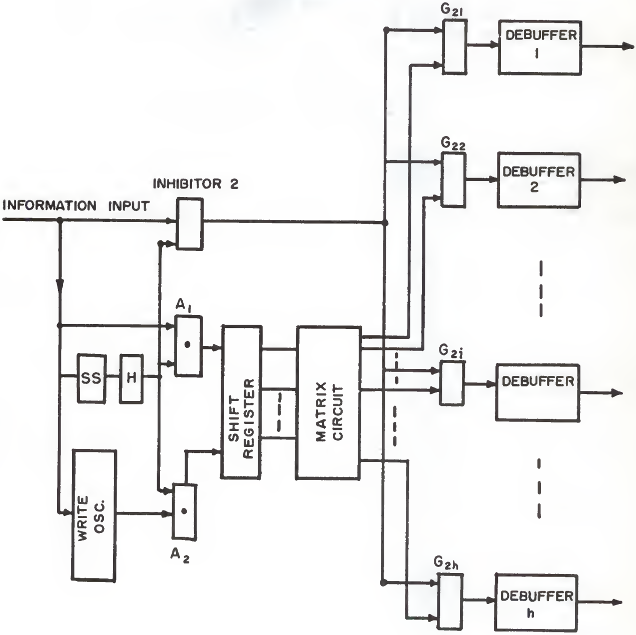


Figure III-2

is converted from serial form into parallel form. The parallel code sets the matrix circuit to activate the proper gate G_i. After the address is read into the shift register the holding circuit deactivates itself which in turn closes the "and gate" A_2 which stops the shift register from reading in any more pulses. The inhibitor 2 is also deactivated allowing the information block to pass to the gates G_{21}, G_{22}, . . . G_{2i}, . . . G_{2h} of which only one has been opened by the matrix circuit. The information is read into the buffer and is read out at the original frequency by a phase locked loop oscillator. The read and write circuits on the buffer stores have been deleted to save confusion.

If this system is to be used for data which occurs at random, it will be necessary to know how large the storage units should be used in order that the probability of loss of information may be made small. Criteria as to how many sources may be handled must be found. Thus, these are the primary problems that will be dealt with in Chapter IV.

The primary advantage of this system is that as long as there is any information in any buffer within the multiplexer system, there will be a continuous flow of information in the transmission line. Therefore, the transmission line is used to its capacity. The system is entirely versatile in that it can be made to handle any input frequency and both probabilistic and continuous data. As stated before, the price paid for such versatility is that a great deal of addressing information must also be transmitted.

CHAPTER IV

A MATHEMATICAL MODEL OF THE VARIABLE FRAME SYSTEM

Introduction

The logic of the digital data dynamic transmission system is such that the incoming messages are broken into blocks, addressed, and merged by a predetermined scheme into a single stream of information. In the analysis that follows it is assumed that the messages are sent whole and that as a message arrives it joins a strict queue. These assumptions are in keeping with the actual multiplexer in that the output stream of data could be sectioned into messages of random length. The only difference is that a message of length t contains information from a single source as it enters the multiplexer, but when the message leaves it may be shortened by a compression factor and may contain information blocks from several sources. The compression factor takes into account the fact that in real time on the average more messages may enter the multiplexer than can be transmitted in real time.

Mathematical Description of the Multiplexer Input

Let the dynamic multiplexer be depicted as is shown in Figure IV-1; there are h inputs each of which is to be represented by a Poisson distribution with different average statistics. There are four basic assumptions which must be made about the inputs if they are to be described by Poisson distributions. In Appendix A it is shown that the i-th input may be represented by a Poisson distribution with average $k_i\tau$ where k_i is the average transition rate, if the following assumptions hold.

THE DYNAMIC MULTIPLEXER

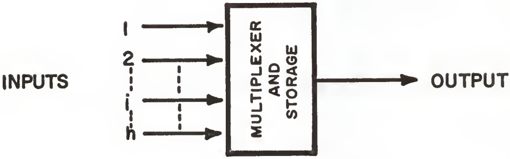


Figure IV-1

i) The probability of a transition during a time interval Δt is assumed to be statistically independent of the number of transitions previous to Δt. A transition is a change of state in the input; a change from "message" to "no message" or vice versa.

ii) The probability of a transition during Δt is assumed to vary as the length of Δt for Δt small.

$$\text{Prob } (\xi_i=1, \Delta t) = P_{\xi_i}(1, \Delta t) = k_i \Delta t$$

where $P_{\xi_i}(1, \Delta t)$ is the probability that the random variable ξ_i equals one transition in the time interval Δt.

iii) The probability of more than one transition in Δt is assumed to be negligibly small, and in fact,

$$\lim_{\Delta t \to 0} \text{Prob } (\xi_i > 1; \Delta t) = 0.$$

Another condition must be placed on each input of the "dynamic multiplexer."

iv) The sources of information feeding the dynamic multiplexer produce messages which are made up of digital pulses, thus a message is quantized in its length. If the duration of an individual pulse is assumed to be very short the quantization can be ignored and the message duration considered as a continuous random variable.

If each of these four conditions holds for each individual input then the i-th input is governed by

$$P_{\xi_i}(n;\ \tau) = \frac{(k_i\tau)^n}{n!}\, e^{-k_i\tau}\ . \qquad \begin{array}{l} n = 0, 1, 2, \ldots \\ i = 1, 2, 3, \ldots h \end{array} \qquad \text{(IV-1)}$$

The function $P_{\xi_i}(n;\ \tau)$ is the probability distribution that the i-th input will make a number of message state transitions in the time interval τ. That is, $P_{\xi_i}(n;\ \tau)$ is the probability that the number of transitions ξ_i is n in τ units of time.

The multiplexer in Figure IV-1 has the following properties: with each input there is associated a large buffer store and as each message arrives at its particular buffer it joins the queue in that buffer. The multiplexer acts as a single server which services each buffer store and transmits information to the single output. The logic of the multiplexer is such that as long as there is any information within any buffer the multiplexer will continue to transmit a constant stream of information to the output.

Collective Input Transitions

Figure IV-2 shows a sample set of inputs to the multiplexer of Figure IV-1. The last time response of Figure IV-2 is a record of all transitions from all h sources. The total number of transitions ξ in the interval τ is

$$\xi = \xi_1 + \xi_2 + \xi_3 + \ldots + \xi_i + \ldots + \xi_h . \qquad \text{(IV-2)}$$

The characteristic function of the random variable ξ is defined as

TYPICAL SET OF INPUTS TO MULTIPLEXER

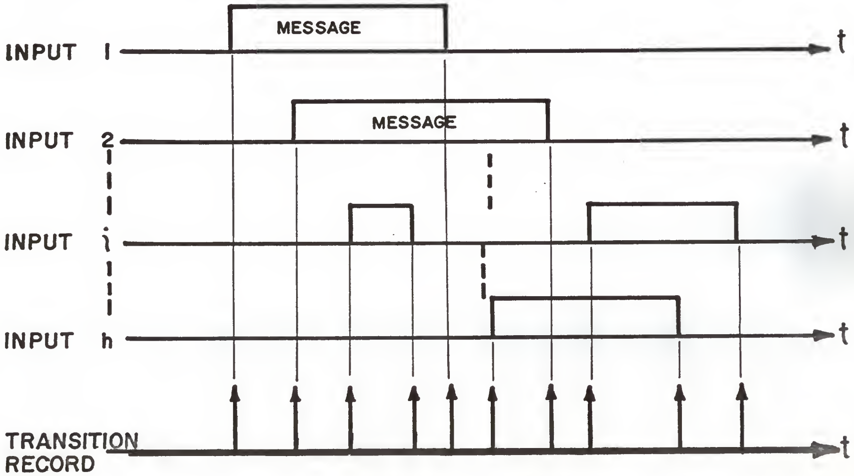


Figure IV-2

$$\Gamma_\xi(s) = E_\xi \langle e^{jsn} \rangle \tag{IV-3}$$

where E_ξ is the expected value operator on ξ, s is a real variable with no physical significance, and ξ is a random variable governing the number of message state transitions as seen collectively by the multiplexer. It can be shown (5) that the characteristic function of the sum of h independent random variables is the product of the h individual characteristic functions.

$$\Gamma_\xi(s) = \Gamma_{\xi_1}(s)\ \Gamma_{\xi_2}(s)\ \Gamma_{\xi_3}(s)\ \ldots\ \Gamma_{\xi_i}(s)\ \ldots\ \Gamma_{\xi_h}(s)$$

The characteristic function of a Poisson distribution is (7)

$$\Gamma_{\xi_i}(s) = e^{-k_i\tau} \sum_{n_i=0}^{\infty} e^{jsn} \frac{(k_i\tau)^{n_i}}{n_i!} = e^{k_i\tau(e^{js}-1)}.$$

The characteristic function of h independent Poisson random variables is

$$\Gamma_\xi(s) = e^{(k_1+k_2+\ldots+k_i+\ldots+k_h)(e^{js}-1)\tau}.$$

Therefore the random variable ξ has a distribution

$$P_\xi(n,\tau) = \frac{[(k_1+k_2+\ldots+k_i+\ldots+k_h)\tau]^n}{n!} e^{-(k_1+k_2+\ldots+k_i+\ldots+k_h)\tau}$$

and if

$$k = k_1 + k_2 + k_3 + \ . \ . \ . + k_i + \ . \ . \ . + k_h \tag{IV-4}$$

then

$$P_\xi(n,\tau) = \frac{(k\tau)^n}{n!} e^{-k\tau}. \tag{IV-5}$$

It is seen that collectively the number of transitions in τ is governed by a Poisson distribution whose parameter k is the sum of the parameters k_i of the individual sources. Thus, the average number of transitions per unit τ that the multiplexer experiences is equal to the sum of the average rates of the individual inputs.

It is now convenient to define the transition from a "no message" state to a "message" state as a "demand" transition and the transition from a "message" state to a "no message" state as an "end" transition. It would seem logical that

$$\xi = \xi_d + \xi_e$$

where ξ_d and ξ_e are discrete random variables describing the total number of demands and ends in τ. By a procedure similar to that used in Appendix A, the demands taken collectively can be shown to be Poisson distributed with average $k_d\tau$ provided that there are an infinite number of sources (9). The requirement of an infinite number of inputs is a direct result of assumption i which states that the probability of a demand during a time interval Δt is assumed to be statistically independent of the number of demands previous to Δt. It is intuitively obvious that the number of demands which can occur within an interval τ is limited by the number of inputs h, for if h is finite it would be very unlikely that more than h demands could occur in a time interval τ which was less than the average message length. Therefore, the assumption of independence between time intervals Δt is not strictly true for any case other than when h approaches infinity.

If the number of inputs h is finite, then the probability distribution governing the number of collective demands would be a function of several variables, namely,

$$P_{\xi_d}(n;\tau) = f(h,k_d,\tau,n)$$

of which the average value would be $k_d\tau$. If the collective demands are assumed to be Poisson distributed with average $k_d\tau$, then

$$P_{\xi_d}(n;\tau) = \frac{(k_d\tau)^n}{n!} e^{-k\tau}.$$

For $n > k_d\tau$ the probabilities given by the Poisson distribution would be an upper bound of the actual probabilities, or

$$\frac{(k_d\tau)^n}{n!} e^{-k\tau} > f(h,k_d,\tau,n) \quad \text{for } n > k_d\tau$$

and if $n < k_d\tau$ the probabilities given by the Poisson distribution would be a lower bound of the actual probabilities, that is

$$\frac{(k_d\tau)^n}{n!} e^{-k\tau} < f(h,k_d,\tau,n) \quad \text{for } n < k_d\tau.$$

It is easy to draw these conclusions knowing that

$$\sum_{n=0}^{\infty} P_{\xi_d}(n;\tau) = 1$$

and that

$$k = k_1 + k_2 + k_3 + \ . \ . \ . + k_i + \ . \ . \ . + k_h.$$

Therefore a system designed to handle collective demands which were assumed to be Poisson distributed, for a finite number of inputs h, would be over-designed. The amount of overcompensation is dependent on how small an integer h is.

Assume that the collective demands are Poisson distributed. This automatically implies that the demands and ends are independent, which in turn implies the ends must also be Poisson distributed. Therefore,

$$P_{\xi_d}(n;\tau) = \frac{(k_d\tau)^n}{n!} e^{-k\tau} \qquad n = 0,1,2,... \tag{IV-6}$$

$$P_{\xi_e}(n;\tau) = \frac{(k_e\tau)^n}{n!} e^{-k\tau} \qquad n = 0,1,2,... \tag{IV-7}$$

Using the characteristic function approach again

$$\Gamma_\xi(s) = \Gamma_{\xi_d}(s)\ \Gamma_{\xi_e}(s)$$

it is seen that

$$e^{k\tau(e^{js}-1)} = e^{(k_d+k_e)(e^{js}-1)\tau}$$

from which

$$k = k_d + k_e. \qquad \text{(IV-8)}$$

Since for every demand transition there is an end transition,

$$E<\xi_d> = E<\xi_e>$$

where E< > is the expected value operator, then

$$k_d = k_e \qquad \text{(IV-9)}$$

Substituting IV-9 into IV-8,

$$k_d = \frac{k}{2}\ . \qquad \text{(IV-10)}$$

Using equations IV-10 and IV-4, equation IV-6 becomes

$$P_{\xi_d}(n;\tau) = \frac{[(k_1+k_2+\ldots+k_i+\ldots+k_h)\frac{\tau}{2}]^n}{n!}\, e^{-(k_1+k_2+\ldots+k_i+\ldots+k_h)\frac{\tau}{2}} \qquad \text{(IV-11)}$$

Single Input-Message Length Distribution

Consider first the i-th source of information. Given a demand transition at some time t_o the probability that the next transition (the end transition) will be found in some time interval t_o+t to t_o+t+dt later, is written as

$$\text{Prob } (t_o+t < \alpha_i < t_o+t+dt) = p_{\alpha_i}(t)dt \qquad \text{(IV-12)}$$

where t is a possible value that the random variable α_i can have and $p_{\alpha_i}(t)$ is the probability density which is sought. Making use of assumption i, it is obvious that

$$\text{Prob } (t_o+t < \alpha_i < t_o+t+dt) = \text{Prob } (\xi_i=0;t) \text{ Prob } (\xi_i=1;dt). \qquad \text{(IV-13)}$$

Now from equation IV-1 (with the proper change in variables)

$$\text{Prob } (\xi_i=0;t) = P_{\xi_i}(0;t) = e^{-k_i t}$$

and from assumption ii

$$\text{Prob }(\xi_i=1;dt) = k_i dt$$

equation IV-12 becomes

$$\text{Prob }(t_o+t < \alpha_i < t_o+t+dt) = k_i e^{-k_i t} dt$$

from which the probability density of message lengths is

$$P_{\alpha_i}(t) = k_i e^{-k_i t}. \tag{IV-14}$$

To show that this density is reasonable it should have an expected value of $\frac{1}{k_i}$.

$$E\langle\alpha_i\rangle = \int_0^\infty t k_i e^{-k_i t} dt$$

Let $k_i t = x$. Then $k_i dt = dx$ and

$$E\langle\alpha_i\rangle = \frac{1}{k_i}\int_0^\infty x e^{-x} dt = \frac{1}{k_i} .$$

Collective Message Length Distribution

Next, the probability density of message length for all inputs taken as a composite is to be found. Let α be the random variable which describes the length of any message taken at random from the composite. Then the probability that the message will have length between t and t+dt is equal to the probability that the message is from input 1 times the probability the length α_1 is between t and t+dt, given the message was from the first input, plus the probability that the message came from the second input times the probability that the message length α_2 is between t and t+dt, given the message was from the second source and so on to the h-th input. This may be written as

$$\text{Prob }(t < \alpha < t+dt) = \sum_{i=1}^{h} \text{Prob(message is from i-th input) Prob}(t < \alpha_i < t+dt \text{ / message is from i}) \tag{IV-15}$$

The probability that a message picked from the composite will be from the i-th input is

$$\text{Prob(message is from the i-th source)} = \frac{\frac{k_i}{2}}{\frac{k}{2}} = \frac{k_i}{k}$$

assuming that the length of the message has nothing to do with the way in which a message is chosen. It then follows from IV-14 and IV-15 that

$$\text{Prob } (t < \alpha < t+dt) = \frac{1}{k} \sum_{i=1}^{h} k_i^2 e^{-k_i t} dt$$

and the composite density is then

$$p_\alpha(t) = \frac{1}{k} \sum_{i=1}^{h} k_i^2 e^{-k_i t}. \tag{IV-16}$$

An equivalent input may now be used in place of the h random inputs. The equivalent input is described by

$$P_{\xi_d}(n;\tau) = \frac{(k_d\tau)^n}{n!} e^{-k_d\tau} \tag{IV-17}$$

$$p_\alpha (t) = \frac{1}{k} \sum_{i=1}^{h} k_i^2 e^{-k_i\tau} \tag{IV-18}$$

and is unique in that the messages may overlap in time. The multiplexer may be thought of as a device which can store and compress the overlapping messages into a real time output.

Description of the Multiplexer Output

In order to calculate the amount of time compression which is necessary the average message length must be found:

$$E\langle\alpha\rangle = \int_0^\infty t p_{\alpha_o}(t)dt = \frac{1}{k} \int_0^\infty t \sum_{i=1}^{h} k_i^2 e^{-k_i\tau} dt$$

$$E\langle\alpha\rangle = \frac{h}{k_o} \tag{IV-19}$$

The average number of demands in τ is

$$k_d \tau = \frac{k}{2} \tau \ . \tag{IV-20}$$

Now from IV-19 and IV-20 the following inequality must hold if message integrity is to be maintained

$$(\frac{h}{k}) \ (k_d \tau) \leq \tau$$

$$\frac{h}{2} \leq 1 \tag{IV-21}$$

and if $\frac{h}{2}$ is greater than 1 then time compression of message length is necessary. Consider the case where h = 2; then the equality holds and time compression is not necessary. This is the time sharing problem where the message rate of the transmission line is equal to the message rate of one of the inputs but because one input uses the line only half of the time, the second input can use the transmission line the other half of the time. Because Poisson distributions were chosen to represent the individual inputs; this implied that each input over a long period of time had information to send half of the averaging time. It would be expected that no more than two inputs could be multiplexed without time compression. Perhaps a more realistic input distribution would be one in which the probability of a long message would be much less than the probability of a long no-message state. This might be accomplished by using a higher ordered Erlang distribution (12).

If $\frac{h}{2} \geq 1$ the compression factor a is given by

$$a \frac{h}{2} = 1$$

$$a = \frac{2}{h} \qquad h = 2,3,4,\ldots$$

Thus the compression factor has values ranging from 0 to 1 excluding h = 1 which is an uninteresting case. Assuming that the multiplexer performs time compression, the distribution of message lengths will be altered at the output to conform to the distribution

$$P_{\alpha_o}(t) = \frac{a}{k_o} \sum_{i=1}^{h} k_i^2 e^{-k_i t} \qquad \text{(IV-22)}$$

where α_o is the random variable which describes the length of the output messages and the expected value is

$$E\langle\alpha_o\rangle = \frac{1}{k_o} . \qquad \text{(IV-23)}$$

The Rates Ratio

Let k_o be the average number of demands appearing at the output of the multiplexer in the time interval τ. Define the rates ratio as

$$\mu \equiv \frac{k_d}{k_o} \qquad \text{(IV-24)}$$

The ratio μ is of considerable importance in queueing theory and is measured in units of erlangs. If $\mu > 1$ then on the average the number of output demands is less than the number of input demands and the queue will continue to grow in length. If $\mu \leq 1$ the queue will grow and shrink as the input fluctuates. A typical value for μ might be 0.8. If $\mu = 1$ the system would be on the verge of instability.

The Expected Value of Queue Length

Consider a storage unit with an input described by:

$$P_{\xi_d}(n;\tau) = \frac{(k_d\tau)^n}{n!} e^{-k_d\tau} \qquad \text{(IV-25)}$$

$$p_\alpha(t) = \frac{1}{k} \sum_{i=1}^{h} k_i^2 e^{-k_i t} \qquad \text{(IV-26)}$$

and an output described by:

$$P_{\xi_o}(n;\tau) = f(n,\alpha_o) \qquad \text{(IV-27)}$$

$$p_{\alpha_o}(t) = \frac{a}{k_o} \sum_{i=1}^{h} k_i^2 e^{-k_i t} \qquad \text{(IV-28)}$$

where a is the compression factor.

Next let time be marked at the "end" transition of some message M_o at the output of the multiplexer, and let this time be $t = 0$. At the instant $t = 0$ there may be a certain number of demands in the buffer store (a demand is the beginning transition of a message). Designate this number of demands in store d_o. If $d_o \neq 0$ then the time $t = 0$ also marks the "demand" transition of a message M_1, which is the next message to be transmitted, and M_1 will begin its service time t_1. If $d_o = 0$ at $t = 0$, the message M_1 will begin its transmission time as soon as it arrives at the multiplexer. The state of the buffer d_1, after M_1 has been transmitted, may be found from the difference equation

$$d_1 = d_o + r_1 - 1 + \delta(d_o) \cdot \qquad \text{(IV-29)}$$

The quantity r_1 is the number of demands which occur at the input during the time it takes to transmit message M_1. The length of the message M_1 is t_1 and the function $\delta(d_o)$ is defined by

$$\delta(d_o) = \begin{cases} 1 \text{ for } d_o = 0 \\ 0 \text{ for } d_o \neq 0 \text{ .} \end{cases} \qquad \text{(IV-30)}$$

It is important to note that the probability that there are n demands received during the transmission of M_1 is given by equation IV-25.

$$P_{r_1}(n;t_1) = \frac{(k_d t_1)^n}{n!} e^{-k_d t_1}$$

The quantities d_1, d_o, r_1, and $\delta(d_o)$ may be considered as range variables or:

$$d_1 = 0,1,2,3,\ldots$$

$$d_o = 0,1,2,3,\ldots$$

$$r_1 = 0,1,2,3,\ldots$$

$$\delta(d_o) = 0,1 \text{ .}$$

Turn now to the case where the q-th message M_q is beginning its transmission time. Let γ_{q-1} be the discrete random variable which describes the number of demands in store at the beginning of the message M_q and let γ_q be the random variable which describes the number of demands in store after M_q has been transmitted. Define ν_q to be the random variable which describes the number of demands which occurred at the input during the transmission of M_q. It follows that

$$\gamma_q = \gamma_{q-1} + \nu_q - 1 + \delta(\gamma_{q-1}). \tag{IV-31}$$

The function $\delta(\gamma_{q-1})$ is defined by

$$\delta(\gamma_{q-1}) = \begin{cases} 1 \text{ for } \gamma_{q-1} = 0 \\ 0 \text{ for } \gamma_{q-1} \neq 0. \end{cases} \tag{IV-32}$$

Solving equation IV-31 for $\delta(\gamma_{q-1})$ and taking the expected value of both sides implies

$$E<\delta(\gamma_{q-1})> = E<\gamma_q> - E<\gamma_{q-1}> - E<\nu_q> + 1. \tag{IV-33}$$

Evaluating the lefthand term of IV-33,

$$E<\delta(\gamma_{q-1})> = (0)\text{Prob } \{\delta(\gamma_{q-1})=0\} + (1)\text{Prob } \{\delta(\gamma_{q-1})=1\}.$$

But from equation IV-32,

$$\text{Prob } \{\delta(\gamma_{q-1})=1\} = \text{Prob } \{\gamma_{q-1}=0\}$$

which implies

$$E<\delta(\gamma_{q-1})> = \text{Prob } (\gamma_{q-1}=0).$$

Assuming that the process is stationary, that is,

$$E<\gamma_q> = E<\gamma_{q-1}>$$

it follows that

$$E<\gamma_q> = E<\gamma_{q-1}> = \text{Prob } (\gamma_q=0)$$

and equation IV-33 becomes

$$\text{Prob } (\gamma_q = 0) \quad = \quad 1 - E\langle \nu_q \rangle \quad = \quad E\langle \delta(\gamma_q) \rangle . \tag{IV-34}$$

It should be noted that the random variable ν_q is the number of demands occurring during the transmission of <u>some</u> message. Therefore,

$$\nu_q \quad = \quad f(\xi_d, \alpha_o)$$

where ξ_d is the number of demands occurring during the transmission of a particular message and α_o is the length of a message picked at random. Therefore

$$E\langle \nu_q \rangle \quad = \quad E_{\xi_d,\alpha} \langle f(\xi_d, \alpha_o) \rangle .$$

First taking the expected value of $f(\xi_d, \alpha_o)$ with respect to ξ_d yields

$$E_{\xi_d} \langle f(\xi_d, \alpha_o) \rangle \quad = \quad \sum_{n=0}^{\infty} n \frac{(k_d \alpha_o)}{n!} e^{-k_d \alpha_o} \quad = \quad k_d \alpha_o . \tag{IV-35}$$

Next take the expected value with respect to α_o to obtain

$$E\langle \nu_q \rangle \quad = \quad k_d E\langle \alpha \rangle \quad = \quad k_d \frac{a}{k_o} \int_0^{\infty} t \sum_{i=1}^{h} k_i^2 e^{-k_i t} dt \tag{IV-36}$$

from which

$$E\langle \nu_q \rangle \quad = \quad \frac{ahk_d}{k_o} \quad = \quad \mu . \tag{IV-37}$$

Substituting equation IV-37 into IV-34, the result is

$$\text{Prob } (\gamma_q = 0) \quad = \quad 1 - \mu . \tag{IV-38}$$

Therefore the probability that the queue length is zero is dependent only upon the value of μ. Squaring equation IV-31 yields

$$\begin{aligned} \gamma_q^2 \quad = \quad & \gamma_{q-1}^2 + 2\gamma_{q-1}\nu_q - 2\gamma_{q-1} + 2\delta(\gamma_{q-1})\gamma_{q-1} + \nu_q^2 - 2\nu_q \\ & + 2\delta(\gamma_{q-1})\nu_q + 1 - 2\delta(\gamma_{q-1}) + \delta^2(\gamma_{q-1}) . \end{aligned} \tag{IV-39}$$

From the definition of $\delta(\gamma_{q-1})$, equation IV-32, it can be shown that

$$2\delta(\gamma_{q-1})\gamma_{q-1} \quad = \quad 0$$

and that

$$\delta^2(\gamma_{q-1}) = \delta(\gamma_{q-1})$$

so that equation IV-39 becomes

$$\gamma_q^2 = \gamma_{q-1}^2 + (\nu_q - 1)^2 + \delta(\gamma_{q-1}) + 2\gamma_{q-1}(\nu_q - 1) + 2\delta(\gamma_{q-1})(\nu_q - 1).$$

Taking the expected value of both sides results in

$$E<\gamma_q^2> = E<\gamma_{q-1}^2> + E<(\nu_q - 1)^2> + E<\delta(\gamma_{q-1})>$$
$$+ 2E<\gamma_{q-1}(\nu_q - 1)> + 2E<\delta(\gamma_{q-1})(\nu_q - 1) . \quad \text{(IV-40)}$$

Assuming stationarity, then

$$E<\gamma_q^2> = E<\gamma_{q-1}^2>$$

and if there is independence between the number of demands in storage and the number of demands arriving then IV-40 becomes

$$0 = E<(\nu_q - 1)^2> + E<\delta(\gamma_{q-1})> + 2E<\gamma_{q-1}>E<\nu_q - 1>$$
$$+ 2E<\delta(\gamma_{q-1})(\nu_q - 1)>. \quad \text{(IV-41)}$$

Now since

$$E<\nu_q> = E<\nu_{q-1}> = \mu$$

and from equation IV-34

$$E<\delta(\gamma_q)> = E<\delta(\gamma_{q-1})> = 1 - \mu$$

equation IV-41 may be written

$$0 = E<\nu_q^2> - 2\mu + 1 + 1 - \mu + 2E<\gamma_{q-1}>(\mu - 1)$$
$$+ 2(\mu - 1)(1 - \mu).$$

Solving for $E<\gamma_{q-1}> = E<\gamma_q>$ the result is

$$E<\gamma_q> = \frac{E<\nu_q^2> - 2\mu^2 + \mu}{2(1 - \mu)} . \quad \text{(IV-42)}$$

The variance operator on the random variable ν_q is defined by

$$V<\nu_q> = E<(\nu_q - E<\nu_q>)^2>$$

which may be simplified to

$$V\langle\nu_q\rangle = E\langle\nu_q^2\rangle - (E\langle\nu_q\rangle)^2$$

from which

$$E\langle\nu_q^2\rangle = V\langle\nu_q\rangle + (E\langle\nu_q\rangle)^2. \tag{IV-43}$$

Remembering that $\nu_q = f(\xi_d,\alpha_o)$ then

$$E_{\xi_d}\langle\nu_q^2\rangle = V_{\xi_d}\langle\nu_q\rangle + (E_{\xi_d}\langle\nu_q\rangle)^2. \tag{IV-44}$$

Now from IV-35

$$E_{\xi_d}\langle\nu_q\rangle = k_d\alpha_o$$

and it can be shown that (2)

$$V_{\xi_d}\langle\nu_q\rangle = k_d\alpha_o.$$

Therefore

$$E_{\xi_d}\langle\nu_q^2\rangle = k_d\alpha_o + k_d^2\alpha_o^2$$

Then taking the expected value with respect to α_o yields

$$E_{\xi_d\alpha_o}\langle\nu_q^2\rangle = k_d E\langle\alpha_o\rangle + k_d^2 E\langle\alpha_o^2\rangle.$$

From IV-43

$$E\langle\alpha_o^2\rangle = V\langle\alpha_o\rangle + \{E\langle\alpha_o\rangle\}^2$$

so that

$$E_{\xi_d\alpha_o}\langle\nu_q^2\rangle = k_d E\langle\alpha_o\rangle + k_d^2 V\langle\alpha_o\rangle + k_d^2\{E\langle\alpha_o\rangle\}^2. \tag{IV-45}$$

Now remembering that

$$E\langle\alpha_o\rangle = \frac{1}{k_o}$$

equation IV-45 becomes

$$E\langle\nu_q^2\rangle = \frac{k_d}{k_o} + k_d^2 V\langle\alpha_o\rangle + \frac{k_d^2}{k_o^2} = \mu + k_d^2 V\langle\alpha_o\rangle + \mu^2. \tag{IV-46}$$

Substituting equation IV-46 back into IV-42 the expected value of queue length is seen to be:

$$E\langle\gamma_q\rangle = \mu + \frac{\mu^2 + k_d^2 V\langle\alpha_o\rangle}{2(1-\mu)} \tag{IV-47}$$

where $V\langle\alpha_o\rangle$ is the variance of the service time distribution given by

$$V\langle\alpha_o\rangle = E\langle\alpha_o^2\rangle - [E\langle\alpha_o\rangle]^2 .$$

The term $E\langle\alpha_o^2\rangle$ will be evaluated first.

$$E\langle\alpha_o^2\rangle = \frac{a}{2k_o}\int_0^\infty t^2 \sum_{i=1}^{h} k_i^2 e^{-k_i t} dt = \frac{a}{2k_o}\sum_{i=1}^{h}\int_0^\infty t^2 k_i^2 e^{-k_i t} dt .$$

Let $x_i = tk_i$; then $dx_i = k_i dt$ and

$$E\langle\alpha_o^2\rangle = \frac{a}{2k_o}\sum_{i=1}^{h}\frac{1}{k_i}\int_0^\infty x_i^2 e^{-x_i} dx_i = \frac{a}{k_o}\sum_{i=1}^{h}\frac{1}{k_i}.$$

The term $E\langle\alpha_o\rangle$ can be obtained from equation IV-19 in the form

$$E\langle\alpha_o\rangle = \frac{1}{k_o}$$

so that

$$V\langle\alpha_o\rangle = \frac{1}{k_o}\sum_{i=1}^{h}\frac{1}{k_i} - \frac{h^2}{k_o^2} = \frac{1}{k_o}\left[\sum_{i=1}^{h}\frac{1}{k_i} - \frac{h^2}{k_o}\right].$$

Substituting into equation IV-47 the result is:

$$E\langle\gamma_q\rangle = \mu\left\{1 + \frac{\mu + k_d\left[\sum_{i=1}^{h}\frac{1}{k_i} - \frac{h}{k_o}\right]}{2(1-\mu)}\right\} \qquad \text{(IV-48)}$$

To show that this equation gives reasonable results assume that

$$k_1 = k_2 \ . \ . \ . = k_i = \ . \ . \ . = k_h$$

then

$$\sum_{i=1}^{h}\frac{1}{k_i} = \frac{h}{k_i}$$

and then it would follow that

$$k_i = \frac{k}{h} = \frac{2kd}{h} .$$

Therefore

$$\sum_{i=1}^{h}\frac{1}{k_2} = \frac{h^2}{2kd} .$$

Then equation IV-48 becomes

$$E<\gamma_q> \;=\; \mu\left[1 + \frac{\mu + h^2(1 - \mu)}{2(1 - \mu)}\right]$$

Let $\mu = 8$ and $h = 30$; then

$$E<\gamma_q> \;=\; .8\left[1 + \frac{.8 + (30)(.2)}{(2)(.2)}\right] \;=\; 14.4$$

But because γ_q must be a discrete number

$$E<\gamma_q> \;\approx\; 15$$

This would say then that out of 30 lines into the multiplexer, all with the same statistics, on the average one half of the inputs would have demands in queue at any particular instant, which would seem quite reasonable.

The Problem in General

At this point in the analysis the problem becomes more clearly defined as to what quantities are needed, and what distributions must be found in order to have meaningful results. It becomes quite evident at this point that the probability distribution for the number of bits stored will be anything but simple, as the number of bits which must be stored is dependent upon the number of messages in queue, the lengths of the messages in queue, as well as the amount of time that a message must wait before it is transmitted.

It is clear that the input messages must be described in a probabilistic manner with respect to demand transitions and length of messages. This may be done on an individual input basis or it may be done on a collective basis. It would be of greater use to describe each source probabilistically and then describe the composite source, but in order to solve the problem for even a simple case it may become necessary to describe the composite input in the most compact way possible. That is, assume that the messages are Poisson distributed in the composite and assume that the lengths of messages are exponentially distributed.

After the input has been properly described the output must be described as far as transition occurrence and message length distribution. The message length distribution for the output is sometimes called the service time distribution in queueing theory. Next, the queue length distribution must be found. The queue length distribution is of course a function of both the input and the output distributions. And in order to write the difference equations for the queue length the following probabilities must be known:

$$\text{Prob } (\xi_d = 1, dt)$$

$$\text{Prob } (\xi_d = 0, dt)$$

$$\text{Prob } (\xi_o = 1, dt)$$

$$\text{Prob } (\xi_o = 0, dt)$$

If both the input and output distributions are known then the above probabilities are also known.

In the analysis carried out in this paper the output demand transition distribution has not been found. The output service time distribution is given by equation IV-28 which is a sum of exponentials. It may be possible to generate the demand transition distribution from the service time distribution but on any account the output transition distribution must be found before any other analysis can be carried out. After the queue length distribution is found the joint distribution between the queue length and bits stored is the next quantity to be found. A procedure to follow might be to find the probability that at some instant of time there are u bits stored given that the queue is d messages in length, times the probability that the queue is d messages in length. It is clear that three kinds of messages may be found in the storage at time t; the first kind is a message which has been partly read into the buffer. The second kind is a message which has been completely read into the store, and the third kind is a message which has been partly read out of the buffer. The probability that a message has been partly read into the buffer is the probability that the message length is greater than $t - (t_o + dt)$ given the message started in the interval dt prior to t_o times the probability that the message started in the interval dt prior to t_o. The other two cases may be similarly stated, and then knowing that the number of bits is proportional to the input frequency a joint distribution between the number of bits stored and the queue length might be found.

CHAPTER V

SUMMARY AND SUGGESTIONS FOR FUTURE STUDY

The type of multiplexing system which should be used in a particular application is dependent upon the kind of data which is to be multiplexed. If the data is of a continuous nature and the pulse repetition rates are nominally submultiples of each other, then a fixed-frame system with pulse stuffing for synchronization is a very efficient method for multiplexing. That is, the amount of synchronizing information which must be transmitted with the multiplexed data is a minimum and the transmission media may be used to its fullest capacity. If the data is of a continuous nature but the inputs are of diverse frequencies a variable frame dynamic multiplexer may be used with the confidence that it is the most efficient multiplexer which can be used. To calculate this system is still a deterministic problem because the input frequencies are all known and the output frequency may then be determined. The amount of storage which is necessary is small and may be calculated knowing the longest amount of time that it will take for the scanner to return to any given line once it has sampled that line.

If the input data is of a probabilistic nature, the dynamic system is the most applicable system of multiplexing. In such a system the inputs must be described in a collective manner or on an individual basis. It is of more use to describe the inputs on an individual basis, as it then becomes possible to calculate the number of inputs which may be multiplexed without a gross overload which cannot be compensated. The inputs must be described probabilistically as to demand arrivals and message length. The output must also be described probabilistically as to processed demands and output message length. The distribution of message length for the output may be taken to be the same as the distribution of message length for the input modified only by a compression factor. The output distribution of demands may then be generated from the output distribution of message length. This, however, has not yet been accomplished in this paper and work is left to be done. Once the output is described the next step is to find the queue length distribution and then a joint distribution of queue length and the number of bits stored. From this distribution it will then be possible to find the distribution of bits stored from which the size of storage may be found.

In order to test this mathematical model it would be advisable to run a Monte Carlo simulation and in this way calculated values may be verified. However, if the input and output distributions become too difficult to handle the computer may have to be used in order to get numerical results. There are other ways to describe the inputs to the multiplexer and one such method is the use of Erlang distributions. These distributions may be useful in that the message length may then have a different average length than the average interval between consecutive transitions. Further study should be devoted to this area.

A SELECTED BIBLIOGRAPHY

1. Filipowsky, R. J., and Scherer, E. H. "Digital data-transmission of the future." IRE Transactions on Communications Systems, vol. CS-9, 88-96, March, 1961.

2. Goode, Harry H., and Machol, Robert E. Systems Engineering. New York: McGraw-Hill, 1957.

3. Hasegawa, T., Tezuka, Y., and Kasahara, Y. "Digital data dynamic transmission systems." IEEE Transactions on Communication Technology, 58-65, September, 1964.

4. Johannes, V. I., and McCullough, R. H. "Multiplexing of asynchronous digital signals using pulse stuffing with added-bit signaling." IEEE Transactions on Communication Technology, vol. com-14, no. 5, 562-568, October, 1966.

5. Lee, Y. W. Statistical Theory of Communication. New York: John Wiley and Sons, 1963.

6. Najjar, H. F., D'Antonio, R. A., and Blasbalg, H. L. "Time division multiplexing of high-speed digital data." 1967 IEEE International Conference on Communication Digest, June, 1967.

7. Papoulis, Athanasios. Probability, Random Variables, and Stochastic Processes. New York: McGraw-Hill, 1965.

8. Ristenbatt, M. P., and Rothschild, D. R. "Asynchronous time multiplexing." 1967 IEEE International Conference on Communication Digest, June, 1967.

9. Ristenbatt, M. P., Daws, J. L., and Lillie, R. L. Asynchronous Time Multiplexing of Speech Sources. Technical Report 156, Cooley Electronics Laboratory, Department of Electrical Engineering, University of Michigan, 1965.

10. Mayo, J. S. "Experimental 224 Mb/s PCM terminals." The Bell System Technical Journal, vol. XLIV, no. 9, 1813-1841, November, 1965.

11. Witt, F. J. "An experimental 224 Mb/s digital multiplexer-demultiplexer using pulse stuffing synchronization." The Bell System Technical Journal, vol. XLIV, no. 9, 1843-1885, November, 1965.

12. Morse, P. M. Queues, Inventories, and Maintenance. New York: John Wiley and Sons, 1958.

APPENDIX A

POISSON DISTRIBUTED TRANSITIONS, i-th INPUT

Consider the ensemble shown in Figure A-1. Each member of the ensemble consists of randomly distributed messages which are represented by blocks; thus there are intervals of time when there is no incoming information and intervals when there is information flow in the i-th input. There are two possible states that the i-th input may be in at a given instant of time, that is a "message" state or a "no message" state. A transition is next defined as the change from a "message" state to a "no message" state or a change from the "no message" state to the "message" state. Of the two types of transitions the change from "no message" to "message" will be referred to as a demand transition, and the demand lasts until the message is terminated by an "end transition."

To show that the transitions may be represented by a Poisson distribution certain assumptions about the transitions must be made.

i) The probability of a transition during a time interval Δt is assumed to be statistically independent of the number of transitions previous to Δt.

ii) The probability of a transition during Δt is assumed to vary as the length of Δt for Δt small. That is, as Δt approaches zero length

$$\lim_{\Delta t \to 0} \text{Prob}\,(\xi_i = 1; \Delta t) = \lim_{\Delta t \to 0} P_{\xi_i}(1; \Delta t) = \lim_{\Delta t \to 0} k_i \Delta t = k_i dt \qquad \text{(A-1)}$$

iii) The probability of more than one transition in Δt is assumed to be negligibly small and in fact

$$\lim_{\Delta t \to 0} \text{Prob}\,(\xi_i > 1; \Delta t) = 0$$

ENSEMBLE OF MESSAGES FROM THE i-th SOURCE

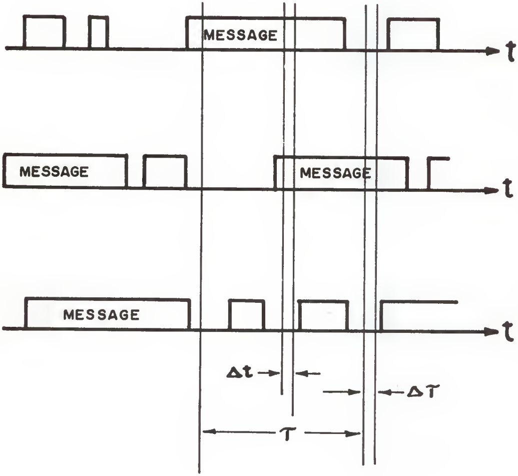


Figure A-I

It is evident from this last assumption that in the limit as Δt approaches zero

$$\lim_{\Delta t\to 0} \text{Prob } (\xi_i=0;\Delta t) + \lim_{\Delta t\to 0} \text{Prob } (\xi_i=1;\Delta t) = 1 \tag{A-2}$$

Consider any interval of time $\tau + \Delta\tau$ in which the probability of no transition may be written as (see Figure A-1):

$$P_{\xi_i}(0;\tau+\Delta\tau) = P_{\xi_i}(0;\tau)\, P_{\xi_i}(0;\Delta\tau) \tag{A-3}$$

where ξ_i is the random variable which describes the number of transitions in any time interval. Substituting A-1 and A-2 into A-3 with the proper change in variable

$$\frac{P_{\xi_i}(0;\tau+\Delta\tau) - P_{\xi_i}(0;\tau)}{\Delta\tau} = -k_i P_{\xi_i}(0;\tau).$$

Taking the limit as Δt approaches zero the following differential equation is generated:

$$\frac{d}{d\tau} P_{\xi_i}(0;\tau) = -k_i P_{\xi_i}(0;\tau) ,$$

the solution of which is

$$P_{\xi_i}(0;\tau) = Ae^{-k_i\tau}.$$

Using the boundary condition

$$P_{\xi_i}(0;0) = \lim_{\Delta\tau\to 0} \text{Prob } (\xi_i=0;\Delta\tau) = 1$$

one obtains

$$P_{\xi_i}(0;\tau) = e^{-k_i\tau}. \tag{A-4}$$

Next consider the probability of n transitions in $\tau + \Delta\tau$.

$$\begin{aligned}\text{Prob } (\xi_i=n;\tau+\Delta\tau) &= \text{Prob } (\xi_i=n-1;\tau \text{ and } \xi_i=1;\Delta\tau)\\ &\quad + \text{Prob } (\xi_i=n;\tau \text{ and } \xi_i=0;\Delta\tau)\end{aligned}$$

Using assumption i the following difference equation may be written:

$$P_{\xi_i}(n;\tau+\Delta\tau) \;=\; P_{\xi_i}(n-1;\tau)\, P_{\xi_i}(1;\Delta\tau) + P_{\xi_i}(n;\tau)\, P_{\xi_i}(0;\Delta\tau) \qquad \text{(A-5)}$$

Substituting A-1 and A-2 into A-5 gives

$$\frac{P_{\xi_i}(n;\tau+\Delta\tau) - P_{\xi_i}(n;\tau)}{\Delta\tau} + k_i P_{\xi_i}(n;\tau) \;=\; k_i P_{\xi_i}(n-1;\tau) .$$

Taking the limit as Δt approaches zero:

$$\frac{d}{d\tau} P_{\xi_i}(n;\tau) + k_i P_{\xi_i}(n;\tau) \;=\; k_i P_{\xi_i}(n-1;\tau) \qquad \text{(A-6)}$$

This differential equation is of the type

$$\frac{dy}{dx} + R(x)y \;=\; S(x)$$

a solution of which is

$$y\, e^{\int R(x)dx} \;=\; s(x)e^{\int R(x)dx}dx .$$

The solution of A-6 is

$$P_{\xi_i}(n;\tau)e^{k_i\tau} \;=\; k_i P_{\xi_i}(n-1;\tau)e^{k_i\tau}d\tau .$$

Now examine the case where n = 1

$$P_{\xi_i}(1;\tau)e^{k_i\tau} \;=\; k_i P_{\xi_i}(0;\tau)e^{k_i\tau}d\tau . \qquad \text{(A-7)}$$

Substituting equation A-4 into A-7 and integrating the right-hand side results in:

$$P_{\xi_i}(1;\tau)e^{k_i\tau} \;=\; k_i\tau + C_1$$

But Prob $(\xi_i=1;0) = 0$ which implies that $C_1 = 0$; therefore

$$P_{\xi_i}(1;\tau) \;=\; k_i e^{-k_i\tau}. \qquad \text{(A-8)}$$

Next let n = 2:

$$P_{\xi_i}(2;\tau)e^{k_i\tau} \;=\; \int k_i P_{\xi_i}(1;\tau)e^{k_i\tau}d\tau \qquad \text{(A-9)}$$

Substituting A-8 into A-9 yields

$$P_{\xi_i}(2;\tau)e^{k_i\tau} = \int k_i\{k_i\tau e^{-k_i\tau}\}e^{k_i\tau}d\tau$$

$$= \frac{(k_i\ )}{2} + C_2 .$$

Now $P_{\xi_i}(2,0) = 0$; thus $C_2 = 0$ and therefore

$$P_{\xi_i}(2;\tau) = \frac{(k_i\tau)^2}{2} e^{-k_i\tau} .$$

The general expression is written after inspection of several values of n and may be proved by induction to be

$$P_{\xi_i}(n;\tau) = \frac{(k_i\tau)^n}{n!} e^{-k_i\tau} \qquad n=0,1,2,... \tag{A-10}$$

ACKNOWLEDGEMENT

The author wishes to express appreciation to Dr. Floyd W. Harris, major professor, for his guidance and encouragement.

The author wishes to express his appreciation for the support given by NASA's Institutional Research Grant NsG-692.

TIME DIVISION MULTIPLEXING OF DIGITAL SIGNALS

by

WAYNE F. WESTGATE

B.S., Kansas State University, 1966

AN ABSTRACT OF A MASTER'S THESIS

submitted in partial fulfillment of the

requirements of the degree

MASTER OF SCIENCE

Department of Electrical Engineering

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1968

ABSTRACT

An investigation of the current methods used to perform time division multiplexing of digital data is summarized in this thesis. A mathematical model for probabilistic data in a variable length frame system is discussed and the problems involved in finding the amount of storage needed are shown in detail. The input data into the variable frame multiplexer is assumed to be given by

$$P_{\xi_d}(n;\tau) = \frac{(k_d\tau)^n}{n!} e^{-k_d\tau}$$

and

$$\bar{p}_\alpha(t) = \frac{1}{k} \sum_{i=1}^{h} k_i^2 e^{-k_i\tau}$$

and an expression for the expected value of queue length is derived.