

May 26, 1964

F. E. FROEHLICH

3,134,962

SERIAL BUFFER

Filed June 8, 1959

8 Sheets-Sheet 1

FIG. 1

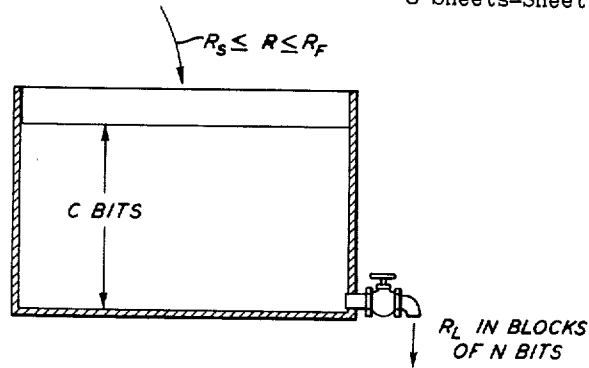
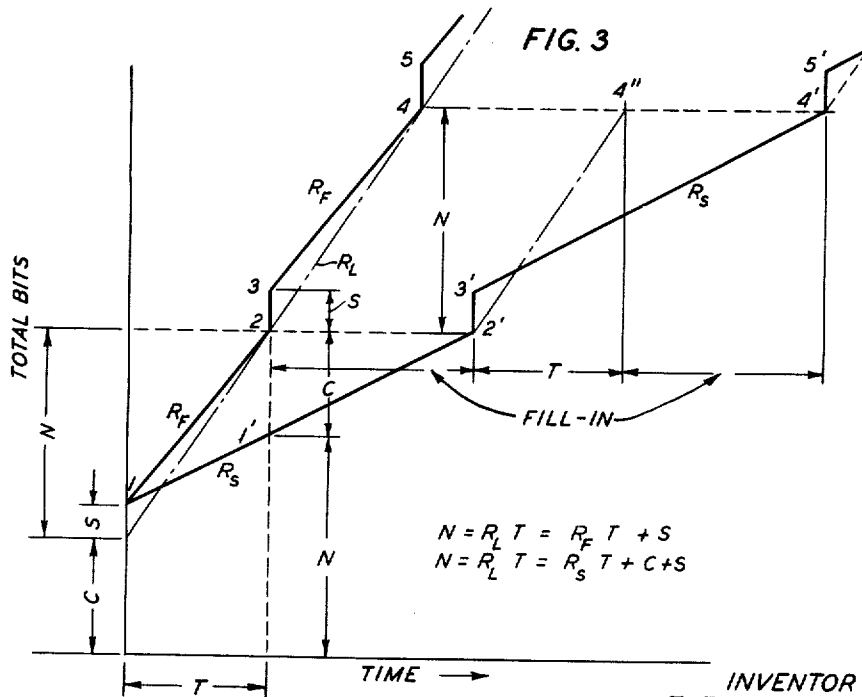
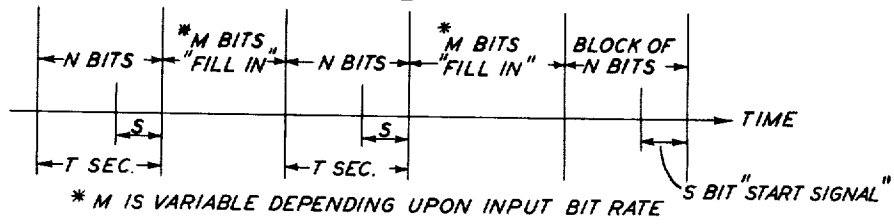


FIG. 2



INVENTOR  
F. E. FROEHLICH

BY *J. D. Kearns Jr.*  
ATTORNEY

May 26, 1964

F. E. FROEHLICH  
SERIAL BUFFER

3,134,962

Filed June 8, 1959

8 Sheets-Sheet 2

FIG. 4

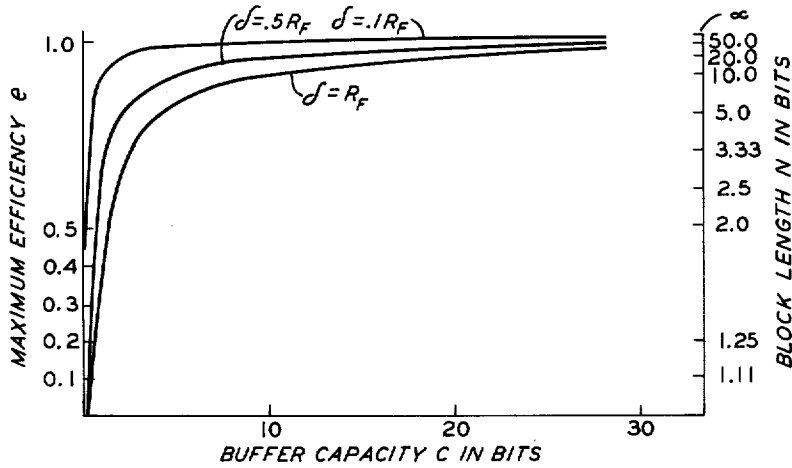


FIG. 5

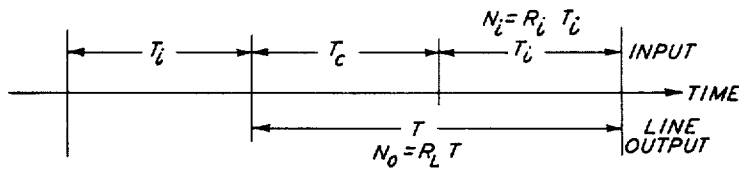
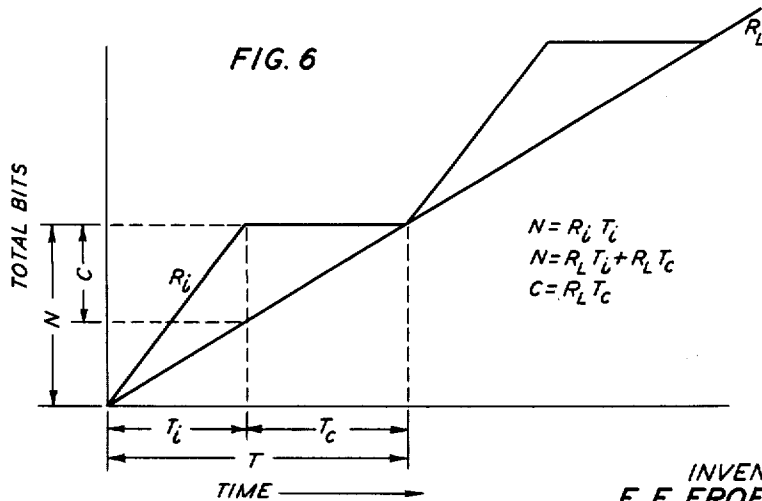


FIG. 6



INVENTOR  
F. E. FROEHLICH

BY *J. P. Kearns, Jr.*  
ATTORNEY

May 26, 1964

F. E. FROELICH

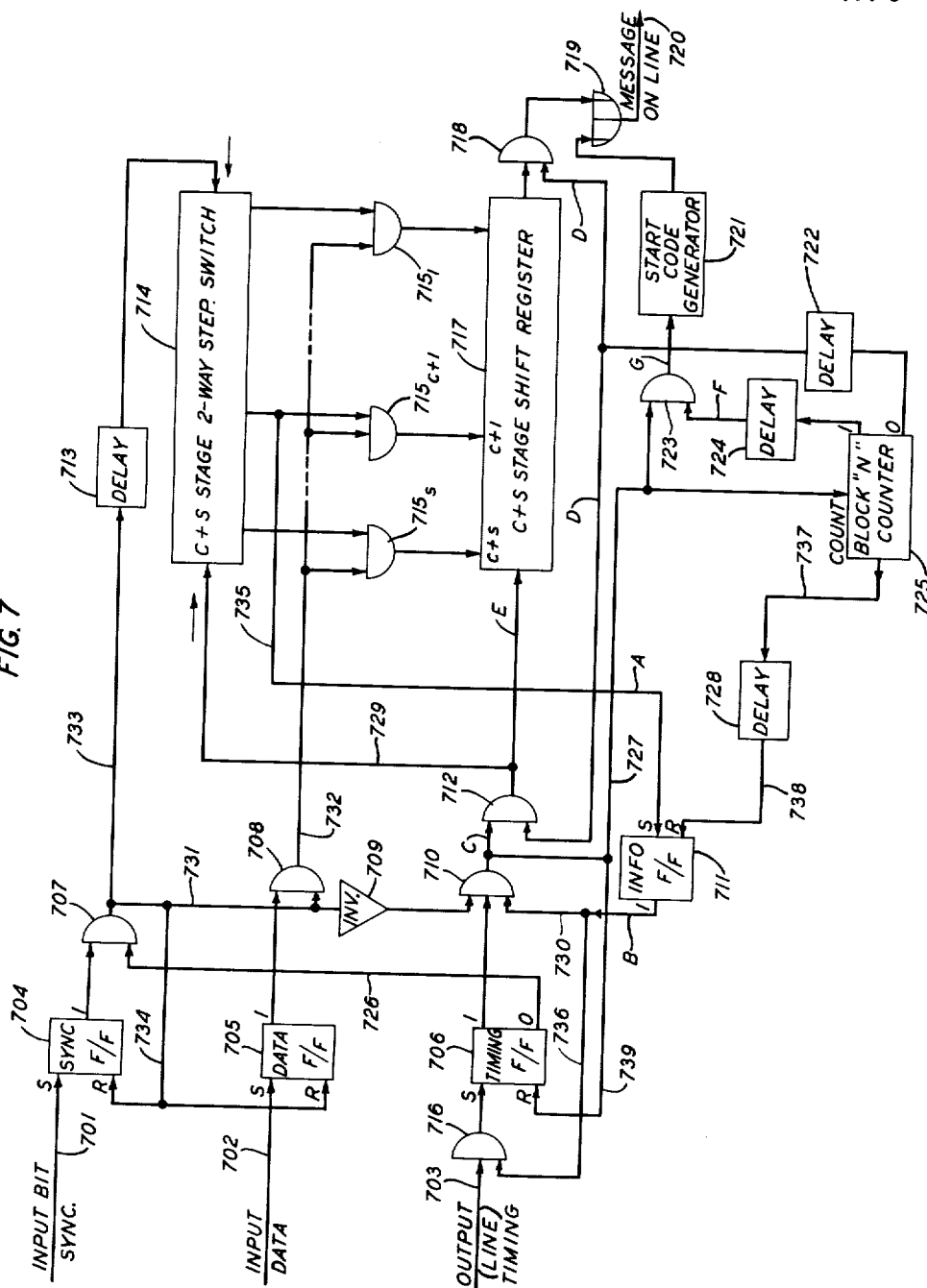
3,134,962

SERIAL BUFFER

Filed June 8, 1959

8 Sheets-Sheet 3

FIG. 7



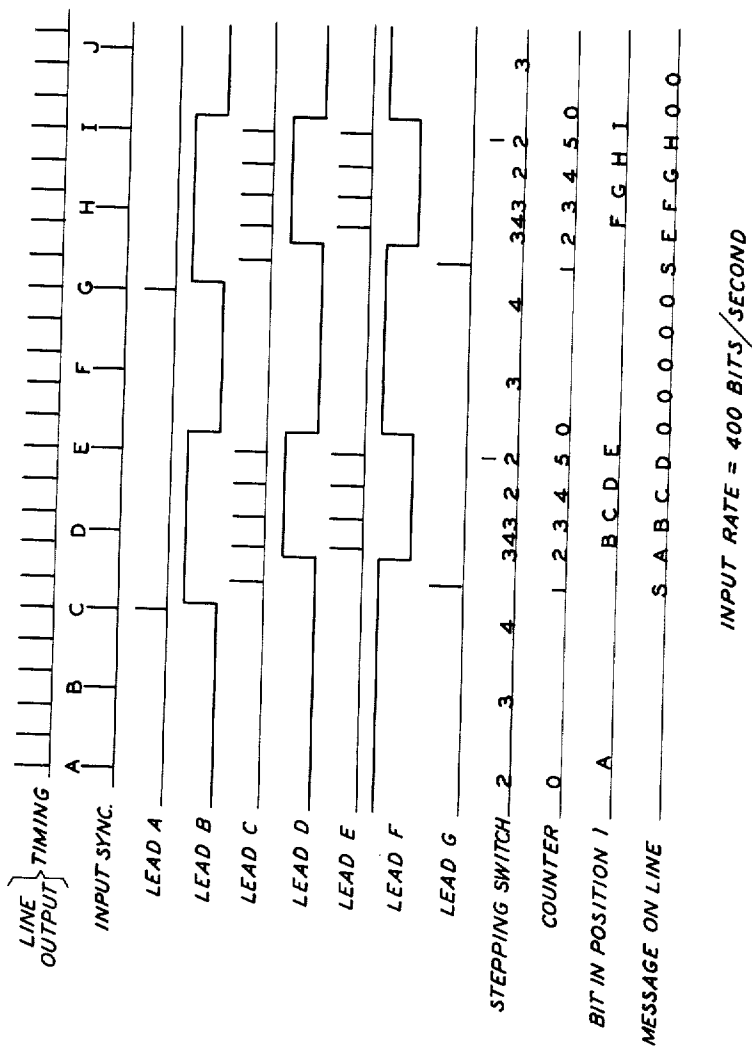
INVENTOR  
F. E. FROELICH  
BY *J. D. Kearns, Jr.*  
ATTORNEY

Filed June 8, 1959

**3,134,962**

8 Sheets-Sheet 4

**FIG. 8**



BY

INVENTOR  
F. E. FROEHLICH

J. P. Kearns, Jr.  
ATTORNEY

May 26, 1964

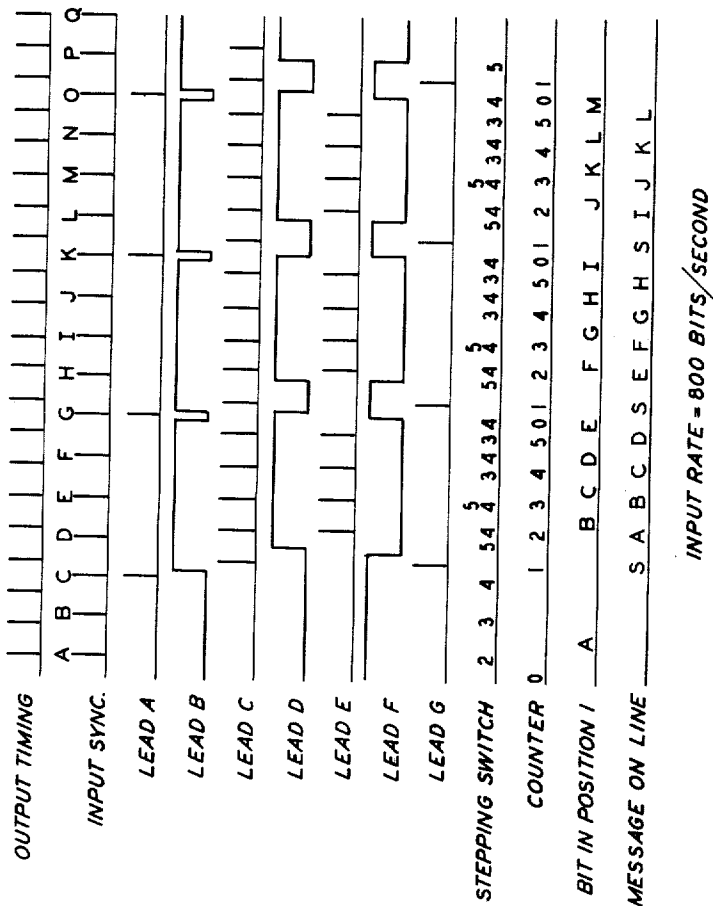
F. E. FROEHLICH  
SERIAL BUFFER

3,134,962

Filed June 8, 1959

8 Sheets-Sheet 5

FIG. 9



INVENTOR  
F. E. FROEHLICH  
BY  
*J. P. Kearns, Jr.*  
ATTORNEY

**May 26, 1964**

F. E. FROELICH

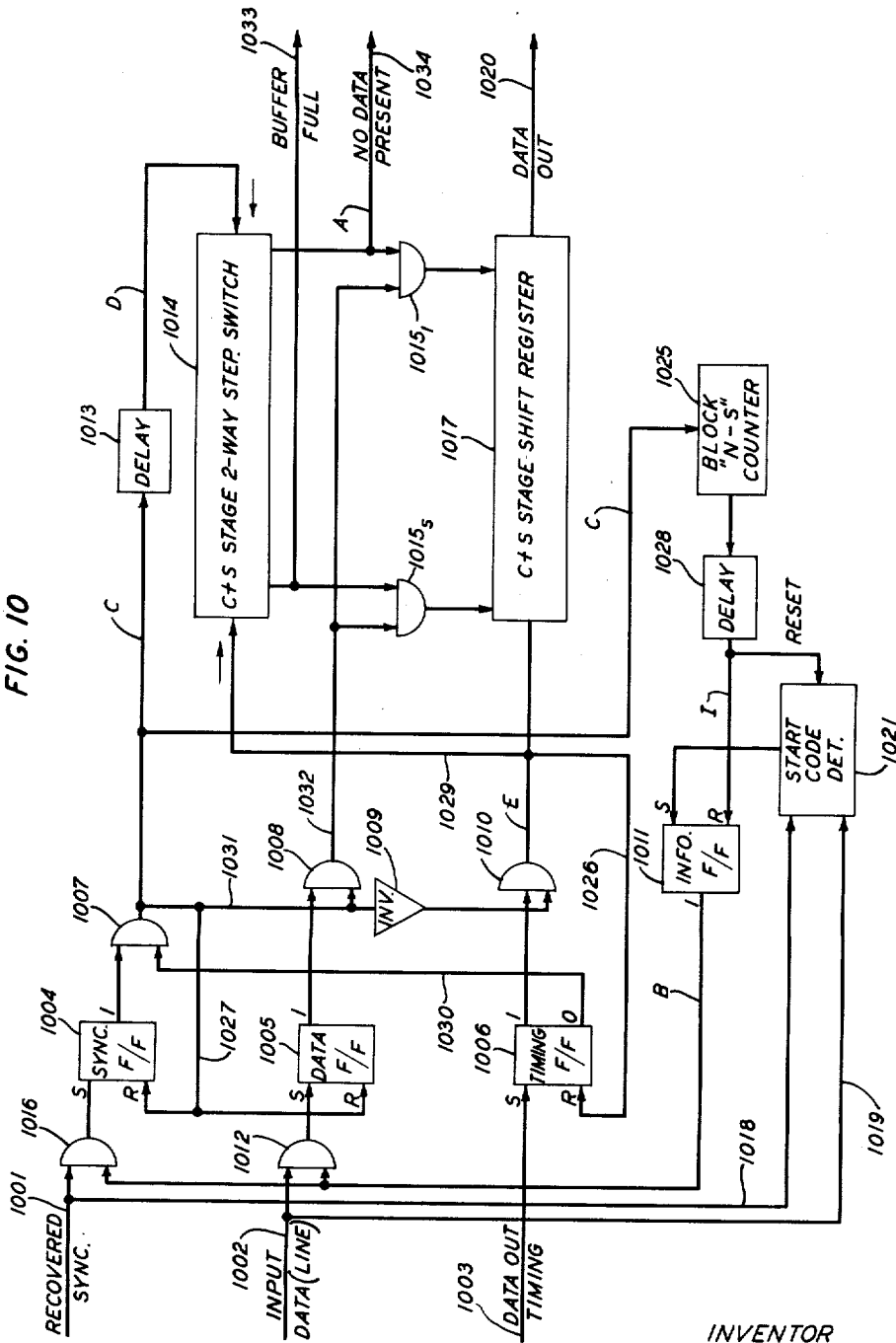
**3,134,962**

SERIAL BUFFER

Filed June 8, 1959

8 Sheets-Sheet 6

**FIG. 10**



INVENTOR  
F. E. FROELICH

BY

*J. P. Kearns, Jr.*  
ATTORNEY



May 26, 1964

F. E. FROEHLICH

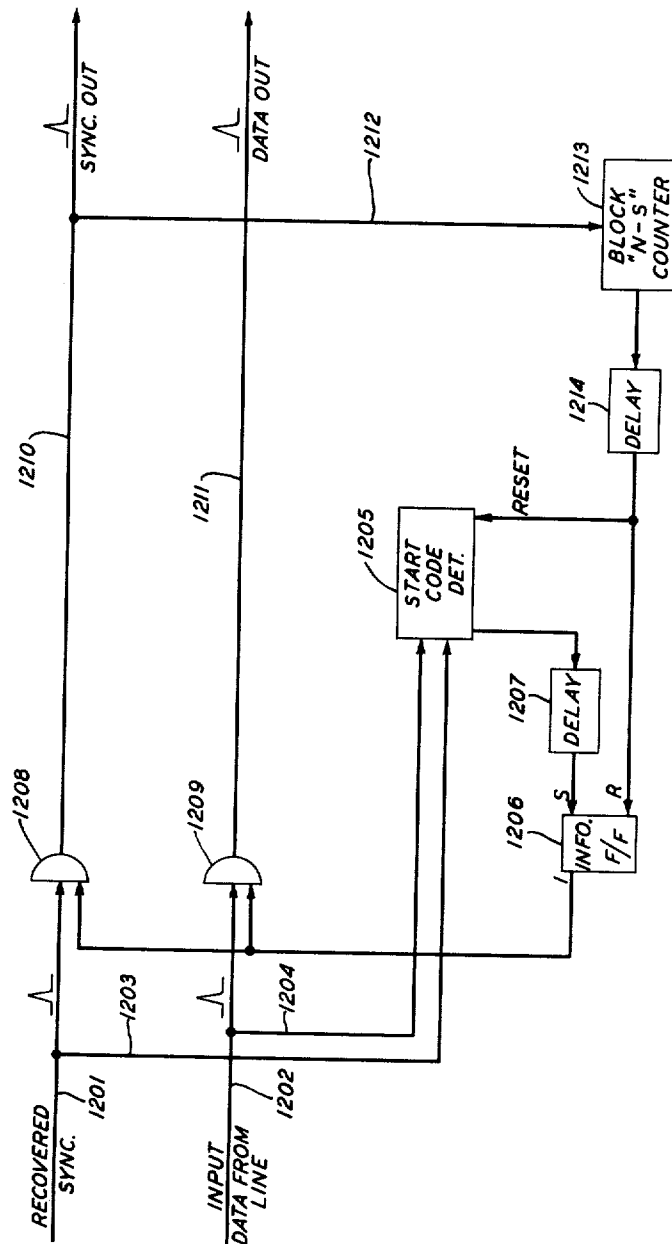
3,134,962

SERIAL BUFFER

Filed June 8, 1959

8 Sheets-Sheet 8

FIG. 12



INVENTOR  
F. E. FROEHLICH  
BY *J. D. Kearns, Jr.*  
ATTORNEY



1

3,134,962

## SERIAL BUFFER

Fritz E. Froehlich, Morristown, N.J., assignor to Bell Telephone Laboratories, Incorporated, New York, N.Y., a corporation of New York

Filed June 8, 1959, Ser. No. 818,868

7 Claims. (Cl. 340—172.5)

This invention relates in general to the transmission of binary, digital data signals in serial form and specifically to the matching of unsynchronized data terminals to a synchronous transmission system.

In recent years attention has been directed more and more to the problem of transmitting intelligence from one geographical location to another in the form of binary digits or bits and insofar as possible using conventional communication media, such as telephone and telegraph lines. Many large businesses which, for example, operate through many branch offices, warehouses, and distribution points are finding it practicable and economical to exercise centralized control of bookkeeping and accounting at a central office through the use of automatic computing equipment into which can be fed on a day-to-day basis directly from the branch offices data on the conduct of that day's business by electrical means. It thus becomes possible to keep up-to-the-minute information continuously at hand on the state of inventories, sales and purchases at a great saving in paperwork and manpower. In short, the problem being considered is that of machines talking to machines.

Fundamental to the handling of intelligence in binary form in synchronization or clocking of the digital information among the various elements in which it is processed, for example in a computer between its input and output equipment. The same synchronization problem is present in the transmission of digital data from one geographical location to another. The art of handling digitalized information has been developing so rapidly of late and the types of equipment which must be integrated in a functioning system have become so diverse that it has not been possible to standardize on a synchronizing bit rate that is common to all digital apparatus. In fact, it may even be undesirable or impossible to adopt such a standard because the different types of apparatus inherently possess different speed characteristics. Nevertheless, ways must be found to interconnect different types of data handling apparatus in order for meaningful systems to be designed.

It has been found that, even though it may be impracticable to adopt a standard clock for all data handling apparatus, it is most advantageous that a data transmission system per se for use with telephone lines, for example, be operated at a fixed standard bit rate between transmitting and receiving station for the greatest transmission efficiency and ease of synchronization. In the particular aspect of the problem stated above then, the solution lies in providing buffering means for matching an unsynchronized source to a synchronized transmission system.

Accordingly, it is a principal object of this invention to make possible the transmission of continuous binary serial messages from asynchronous data sources over synchronous transmission systems.

It is another object of this invention to improve the matching of fluctuating data source generation rates to a fixed transmission rate in a data transmission system.

It is a further object of this invention to simplify the synchronization of transmitter and receiver in a high-speed digital data transmission system.

It is a still further object of this invention to enable

2

the transmission of continuous serial messages occurring at one bit rate at a higher bit rate.

According to this invention, a buffer is inserted between a data source having a fluctuating asynchronous bit rate and a transmission system having a fixed synchronous bit rate. The buffer continuously accepts data at the varying source rate, stores it temporarily, and releases it in blocks of fixed length under the control of a synchronous clock whenever its capacity has been reached. The relations among the range over which the fluctuation of the source bit rate may be allowed to vary, the capacity of the buffer, the length of the transmitted block, and the synchronous transmission rate are such that the capacity of the buffer is only a fraction of the transmitted block length. Between transmitted blocks of characters no output appears on the line, but the transmitting clock continues operating so that the next block of characters starts in the proper phase. Synchronization information may thus be readily recovered at the receiver.

A similar buffer further according to this invention is inserted between the synchronized receiver and a suitable data sink adapted to accept information at some rate lower than the synchronous rate but at least slightly higher than the average rate at which information was delivered to the transmitting buffer.

A particular advantage of this invention is that no constraint need be put on the form of the message to be transmitted. No indexing or framing signals are required in the original message. In addition, no sequences of bits or special character codes appearing in the original message are prohibited. Only a start signal is necessary for each transmitted block, but this is inserted by the buffer itself at the transmitting end of a system and removed at the receiving end. The originator and recipient of the message need not be aware of the presence of this start signal, because it is suppressed before the message is delivered to the sink.

The relatively small capacity required by a buffer according to this invention is a further advantage. The capacity of the buffer relative to the transmitted block length can be designed to approach arbitrarily close to any transmission efficiency desired.

This invention and its several objects and advantages will be more completely understood and appreciated after a consideration of the following detailed description together with the drawing, in which:

FIG. 1 is a diagrammatic analogy between the principle of this invention and certain hydraulic systems;

FIG. 2 shows the structure of a transmitted message required for a system embodying principles of the invention;

FIG. 3 is a rate diagram showing the relation between input rates and transmitting rates for a buffer storage system according to this invention;

FIG. 4 is a graph relating buffer storage capacity to block length and transmission efficiency for several input bit rate fluctuations;

FIG. 5 shows a message structure for the case where the input bit rate exceeds the transmission rate for short intervals of time;

FIG. 6 is a rate diagram showing the relation between an input bit rate and the transmitting rate where the former exceeds the latter;

FIG. 7 is a block diagram illustrative of a buffered data transmitter according to this invention;

FIGS. 8 and 9 are pulse timing diagrams of aid in explaining the operation of the transmitter shown in FIG. 7 at a slow and a fast input rate, respectively;

FIG. 10 is a block diagram illustrative of a buffered data receiver according to this invention.

FIG. 11 is a pulse timing diagram of aid in explaining the operation of the receiver shown in FIG. 10; and

FIG. 12 is a block diagram of a simplified receiver according to this invention.

Referring now to FIG. 1, a homely analogy is seen to make the principle of this invention quite clear. Pictorially the buffer of this invention can be represented by a supply tank for water having a storage capacity  $C$ . Water is continuously being added to the tank at a volume rate  $R$ , lying between an upper limit  $R_F$  and a lower limit  $R_S$ , over which no specific control is exercised. Now an attendant is employed to watch the liquid level in the tank. Whenever this level reaches the capacity  $C$  and before the tank overflows, he opens the drain cock and holds it open for a predetermined fixed length of time. Assuming that the difference in pressure head between full and empty conditions negligible, the liquid flows from the drain at a constant rate  $R_L$ .

If, for example, the filling rate fluctuates between 500 and 900 gallons per hour and the withdrawal rate is 1000 gallons per hour, a tank of 250 gallons capacity would be emptied in one half-hour and would be refilled in another half-hour at the lowest filling rate of 500 gallons per hour. Five hundred gallons are withdrawn each time the drain is opened regardless of the filling rate. At the highest filling rate, however, the tank level falls by only 50 gallons in a half-hour and refills to capacity in 3½ minutes.

An efficiency can be assigned to the storage tank as a percentage of the total time withdrawals are being made. On this basis the efficiency of the system varies from 50 percent at the slowest rate to about 89 percent at the fastest rate. This is accomplished without imposing any constraint on the manner in which the filling rate fluctuates between the assigned fastest and slowest rates. In this example the capacity of the tank is only one-half that of the volume of liquid withdrawn on each operation of the drain.

A further analogy can be drawn to the start-stop teletypewriter system. In this system each message character is comprised of a fixed number of bits. The transmission rate for an individual character is fixed by the speed of the distributor. When the distributor operates the input rate must equal the transmission rate, but the overall transmission rate of an entire message is determined by the interval between characters. If each character is delivered to the transmitter in parallel form, storage capacity equal to the fixed number of bits in a message character must be provided until that character has been transmitted in serial form. As between a transmitter and a receiver, synchronization is necessary only during the interval in which the distributor is rotating. In a transmission system with which the present invention is intended to operate, however, continuous synchronization is necessary and this is not possible in the teletypewriter system. Furthermore, in teletypewriter systems no variation between the input rate and the transmission rate is permissible within a character.

The crux of this invention lies in a method for matching a non-synchronous data source to a synchronous transmission system by continuously storing information bits at the fluctuating source rate and delivering them to the transmission system in uniform blocks at the synchronous rate. In order to accomplish this a certain transmitted message structure must be devised. Assuming that the data source rate never exceeds the transmission rate, although it may vary in any manner between the latter rate and a set lower rate, a message structure as shown in FIG. 2 may be formulated. Let the message train comprise a sequence of exactly  $N-S$  information bits. Each block of  $N$  bits begins with a particularly chosen start signal consisting of  $S$  bits. Following the block of  $N$  bits are what may be regarded as "fill-in" bits. The number of these fill-in bits varies in accordance with the momentary input rate of the data source. These

fill-in bits may be actually transmitted or not as necessary to maintain synchronism at the receiving end of the transmission system. If no changes in output are actually transmitted the fill-in bits may be regarded as a series of zeros.

At the receiving end of the system synchronization recovery is had from the message block and fill-in bits. Further knowing the structure of the start signal and the number of information bits in each block, the receiver is able to decode the message.

In order to practice this invention a storage buffer is necessary. The properties requisite to such a buffer are derived from the following considerations. Let  $R_F$  and  $R_S$  be the fastest and slowest information bit rates, respectively, of the data source which a given system is expected to handle. Further, let  $R_L$  be the synchronous information rate on the transmission medium with the constraint that  $R_L$  is greater than  $R_F$ . Now define the total fluctuation  $\delta$  in the input rates, that is,

$$\delta = R_F - R_S \quad (1)$$

The maximum transmission efficiency  $e_m$  (or minimum redundancy) of the system that may be achieved occurs for a continuous input rate of  $R_F$ , or

$$e_{\max} = \frac{R_L}{R_F} \quad (2)$$

The minimum transmission efficiency correspondingly occurs for a continuous input rate of  $R_S$ , or

$$e_{\min} = \frac{R_L}{R_S} \quad (3)$$

If  $C$  is defined as the storage capacity required of the buffer, it is possible to derive a relationship among  $C$ ,  $e$ , and  $\delta$ .

One requirement of a buffer for a message train of the structure shown in FIG. 2 is that during the transmission of a block of  $N$  bits, the buffer shall neither overflow nor completely empty. A graph showing the operation of the system is shown in FIG. 3. The broken line represents information flowing out of the buffer at a rate  $R_L$ . The slopes of the solid lines represent the fastest and slowest input rates to the buffer  $R_F$  and  $R_S$ , respectively. From this graph the following equations for the system can be deduced.

Since  $N$  bits are always transmitted at a rate  $R_L$ , the constraint for the fastest input rate  $R_F$  is

$$R_L T = R_F T + S = N \quad (4)$$

where  $R_L$  and  $R_F$  are the output and the fastest input rates, respectively,  $S$  is the number of bits in the start signal, and  $T$  is the time required to transmit  $N$  bits at the transmission rate  $R_L$ .

Similarly, the constraint for the slowest input rate  $R_S$  is

$$R_L T = R_S T + C + S = N \quad (5)$$

where  $C$  is the effective storage capacity of the buffer in bits, not including a reserve for start bits.

The required capacity  $C$  may be solved for by combining Equations 4 and 5. Thus,

$$C = \frac{S(R_F - R_S)}{R_L - R_F} \quad (6)$$

Equation 6 shows how the effective buffer storage capacity is related to the data source and line bit rates.

FIG. 3 is a graphic showing of the relation between the fastest and slowest permissible input rates and the transmission rate of a buffer storage system according to this invention and is further illustrative of Equations 4 through 6. Time is represented by the abscissa and total bits added to the buffer from an initially empty state, by the ordinate. An initially empty buffer fills at any input rate to point 1 on the graph. At this time a block of  $N$  message bits, including the  $S$  bit start signal, is transmitted at the line rate  $R_L$  in time  $T$ . While the block  $N$  is being transmitted, input bits are being continuously delivered

5

to the buffer. If the input rate is constant at the fastest rate  $R_F$ , the total number of bits delivered to the buffer is the number corresponding to the location of point 2 which is equal to  $C$  plus  $N$  bits. Thus, the buffer has just refilled at the time the last of the first block of  $N$  bits has been transmitted. A new start signal is immediately generated corresponding to filling the buffer to point 3. A second block  $N$  is transmitted. More input bits are added, and at the end of the second block  $N$  the point 4 is reached. A further start signal is generated, represented by the interval between points 4 and 5. This action continues indefinitely until an entire message has been transmitted. It is seen that the buffer is being continuously filled, and message bits are being delivered to the line in blocks of  $N$  bits prefaced only by the start signal. No fill-in between blocks is required.

At the slowest input rate  $R_S$  which determines the relation of the transmitted block length to the capacity of the buffer the transmitted block begins as before at the time the buffer contains  $C$  bits. A start signal proceeds first, and is followed by the remainder of the block  $N$ . At point 1 on FIG. 3 the buffer gains new input bits at the slowest rate  $R_S$  but the buffer empties in time  $T$  just as the  $N$ th bit is transmitted as indicated by point 1' in FIG. 3. Bits continue to be added to the buffer at the rate  $R_S$  until point 2' is reached. Now the same total number of bits at the rate  $R_S$  has been added to the buffer from its initially empty condition as was added at the rate  $R_F$  to reach point 2. The time interval between points 2 and 2' is due to the difference between the fastest and slowest bit rates. This time interval may be considered to be occupied by fill-in bits at the line rate  $R_L$ . At the point 2' the buffer is again full and the transmission of the second block of  $N$  bits proceeds for a new time interval  $T$  to point 4''. The distance between points 2' and 3' represents the insertion of the start signal bits which are transmitted in addition to the message bits proper. Meanwhile, as before, the buffer continues to fill after time  $T$  at rate  $R_S$  until it is again full at point 4'. More fill-in is required.

At any input rate between  $R_F$  and  $R_S$  a sequence similar to that described for the slowest rate  $R_S$  transpires with the difference that the fill-in intervals are shorter in duration. As the input rate fluctuates the fill-in intervals are adjusted accordingly.

From the above equations the following more useful expressions in terms of constants of the system, rather than particular bit rates, can be derived.

Substitution of Equations 1, 2 and 3 into Equation 4 yields

$$C = \frac{\delta S}{R_L(1 - e_{\max})} \quad (7)$$

or

$$C = \frac{(e_{\max} - e_{\min})S}{1 - e_{\max}} \quad (8)$$

It may be noted from Equations 2 and 4 that the length  $N$  of a transmitted block of message is completely defined by  $e_{\max}$  and  $S$ , that is

$$N = \frac{S}{1 - e_{\max}} \quad (9)$$

FIG. 4 constitutes a family of curves based on the numerical evaluation of the above Equations 8 and 9. These curves graphically show the relation between  $e_{\max}$ ,  $\delta$  and  $C$  on the assumption that the number of start bits  $S$  equals unity. It can be seen that the curves rise rapidly from a buffer capacity of zero to approach an efficiency of 75 percent or more with only four or five bits storage capacity, regardless of the degree of fluctuation of input bit rates. Thus, for a storage capacity of ten bits a block length exceeding 50 bits can be accommodated at a maximum efficiency approaching 100 percent for  $\delta=0.1R_F$  and over 90 percent for  $\delta=R_F$ . The block length for a 75

6

desired efficiency can also readily be determined from FIG. 4 in an obvious manner.

A few specific examples of the size of the buffer may be given. For the special case in which  $R_S=0$ , that is  $\delta$  in Equations 1 and 7 equals  $R_F$ , the capacity of the buffer must be  $N-S$ . In this case evidently the buffer capacity is approximately equal to the block length. In all other cases  $C$  is smaller than  $N$ .

For  $e_{\max}=0.98$  and  $e_{\min}=0.8$ ,  $C=9$  from Equation 8. A buffer store of 9 bits capacity can accept fluctuations in the data source rate between 800 and 900 bits per second if the transmission rate is 1000 bits per second. According to Equation 9 a block length of 50 bits can be handled. Similarly, a buffer store of 5 bits capacity will handle any input variation between 400 and 900 bits per second for a block length of 10 bits, at the same transmission rate.

The buffer store described herein is not of course limited in use to the transmission of messages in blocks of fixed length as shown in FIG. 2. Such a buffer may be used for continuous transmission until the buffer empties. At this time an "end-of-block" signal would be required to indicate to the receiver that it is necessary to wait until a new "start-of-block" signal is received. Although this approach makes possible a maximum transmission efficiency of 100 percent, there are disadvantages in that provisions must be made to generate both start and end signals at the transmitter and to recognize both start and end signals at the receiver. The use of such an end signal may limit the code combinations permissible in the original message. The system proposed here, using the uniform message block, eliminates the end-of-block signal and requires only the start-of-block signal, thereby placing no restriction whatsoever on the code combinations permissible in the original message.

A useful extension of the "uniform message block" buffer store method is found in its adaptation to the case where the data source rate exceeds the transmission rate for short time intervals. Many computers are designed to transfer small blocks of information at very high speeds to magnetic and paper tapes and the like. It might be possible to connect such a computer to a telephone line, provided that the buffer is capable of accepting these blocks of information at high speed.

A message structure such as shown in FIG. 5 can be handled. The input rate  $R_i$  exceeds the line rate  $R_L$ . A number of input bits  $N_i$  exceeding the capacity  $C$  of the buffer can be accepted provided a number of bits  $N_o$  are immediately transmitted at the line rate  $R_L$ . Input bits can be accepted for a time interval  $T_i$  provided a waiting period of at least  $T_c$  is established between input blocks. The number of bits  $N_o$  transmitted in a block can then equal the number of bits  $N_i$  from the source, and a time  $T$  exceeding  $T_i$  is required in which to transmit them. The difference  $T_c$  between  $T$  and  $T_i$  is the time which must be bridged by the bits stored in the buffer.

FIG. 6 shows this more clearly. The number of bits in the input block equals the number in the transmitted block, i.e.,  $N_i=N_o$ . The buffer capacity must be at least equal to the difference between the total number of bits in the block and the number that can be transmitted at the rate  $R_L$  in the time  $T_i$ . The number of bits  $C$  stored in the buffer during time  $T_i$  are then transmitted in time  $T_c$  immediately following a number of bits equal to the difference between  $N$  and  $C$ . The equations shown on FIG. 6 are obtained by inspection from a study of the graph and correspond in kind to those considered in connection with FIG. 3 above. An equation for the maximum size  $N$  of the input block that can be handled by a given buffer of capacity  $C$  can be derived from the equations on FIG. 6 and is

$$N = \frac{C}{1 - \frac{R_L}{R_i}} \quad (10)$$

The message structure of FIG. 5 and the transmission diagram of FIG. 6 represent an ideal condition and assume that the original message blocks and the capacity of the buffer match so that transmission is continuous and no start signals are required. As this would rarely be the case in practice, intervals of varying length can be expected between input blocks. Therefore, in general a start signal is necessary at the beginning of each transmitted block. Fill-in bits are then inserted between successive transmitted blocks of information.

An illustrative embodiment of a transmitting terminal employing the principles of this invention as outlined above is shown in block form in FIG. 7. Assume that each input bit, which may be either a binary "1" or "0" according to the content of the message, is accompanied by an input bit synchronization signal, which is invariably composed of a binary "1's," and that an output timing signal at the precise rate of  $R_L$  pulses per second is available at all times. The buffer must be able to deliver bits to the line in block form as shown in FIG. 2 at the exact rate of  $R_L$  bits per second.

The buffer proper comprises a shift register 717 having a storage capacity of C plus S bits and a two-way stepping switch 714 having C plus S positions. Interconnecting individual storage cells of the shift register 717 and the several positions of the stepping switch 714 are a plurality of AND-gates 715 through 715<sub>S</sub>. The remainder of the diagram comprises a start code generator 721, a block counter 725 having a capacity N, and access logic circuitry as shown.

Shift register 717 may be of any well known type employing electron tubes, transistors, relays, magnetic cores, or the like. Stepping switch 714 may similarly employ a stepping electron tube of special multielectrode construction, a stepping relay steering circuit, or a chain of bistable flip-flops. The only special requisite is that bi-directional shifting be possible and that only one output at a time be activated. The sync 704, data 705, timing 706 and information 711 flip-flop circuits may be conventional electron tube or transistor bistable circuits. Counter 725 may be any type of dividing counter which produces an output after a fixed number of input pulses.

The operation of the transmitting buffer shown in FIG. 7 is based on the enabling of one only of the AND-gates 715 at a given instant of time under the control of the stepping switch 714. For purposes of simpler description assume that the register 717 is empty and that the AND-gate 715<sub>1</sub> is the only one enabled. Information flip-flop 711 and timing flip-flop 706 are in the reset condition, thereby inhibiting output timing pulses on line 703 from reaching register 717. The first input bit arrives on line 702 and, if it is a binary "1," sets data flip-flop 705. Simultaneously a sync bit arrives on line 701 to set sync flip-flop 704. Inasmuch as timing flip-flop 706 has been assumed to be in the reset condition, there is an enabling output on the lead marked "0." Therefore, AND-gate 707 has two enabling inputs and thus produces an output on lead 731 which connects to AND-gate 708. Assuming that the first data bit is a binary "1," AND-gate 708 also has an enabling input from data flip-flop 705, thereby producing an output on lead 732 which connects to the inputs of AND-gates 715. Only AND-gate 715<sub>1</sub> is enabled by stepping switch 714 and therefore a binary "1" is delivered to the first position of register 717, which thereupon stores it. Had the first data bit been a binary "0" flip-flop 705 would not have been set, and a binary "0" would be stored in the first position of register 717. Switch 714, however, advances for each input sync bit.

The output from AND-gate 707 is also connected by way of lead 733 and delay element 713 to the right-hand end of stepping switch 714, as well as by lead 734 to the reset inputs of sync flip-flop 704 and data flip-flop 705 and further by lead 731 to inverter 709. Data 705 and sync 704 flip-flops are immediately reset in preparation for the receipt of the next input bit. Meanwhile the delayed

sync bit passing through element 713 is delivered to the stepping switch 714 to enable the next AND-gate 715<sub>2</sub>. The next succeeding input bit is routed to the second storage cell in register 717, and the accompanying sync bit steps switch 714 another position to the left. Additional input bits are routed in succession to other storage cells in register 717 until AND-gate 715<sub>C+1</sub> is enabled on the next input bit after storage cell C is filled. Other functions of the sync impulses on leads 731 and 734 are considered below.

The sync impulse that enables gate 715<sub>C+1</sub> is also applied over lead 735, also designated A, to the set input of information flip-flop 711. The latter flip-flop produces a sustained output on lead B, which by way of lead 736 enables AND-gate 716 and by way of lead 730 enables three-input AND-gate 710. Upon the occurrence of the next output timing pulse on lead 703, timing flip-flop 706 is set to produce a sustained output on its lead "1" which connects to another input of AND-gate 710. Gate 710 now has at least two of its inputs enabled, and, provided sync flip-flop 704 has been reset, is fully enabled. The resulting output on lead C from gate 710 resets timing flip-flop 706 on lead 739 and simultaneously over lead 727 enables counter 725 and partially enables AND-gate 723.

Counter 725 in its normal off state has a sustained output on the lead marked "1." This output is connected through delay element 724 and lead F to AND-gate 723. At the time gate 710 is enabled, as previously described, gate 723 is therefore fully operative and an output appears on lead G to actuate start-code generator 721. This generator may be a monostable multivibrator delivering a single output pulse for every input pulse which serves as an indexing pulse for a transmitted pulse train. It may be more or less elaborate depending on how many pulses are needed to give an unambiguous start signal on a given transmission system. The output of this generator is delivered to outgoing line 720 through OR-gate 719.

Counter 725 counts the pulses in the start code and then changes the output marked "0" to "1." The latter output is transmitted to lead D through delay element 722 and AND-gate 718 is enabled. The output on lead D also effects a sustained enabling output to AND-gate 712. After the counter output on lead F vanishes gate 723 is disabled and the start code generator is stopped. Assuming that the start signal is a single impulse, the next output timing pulse on line 703 again sets timing flip-flop 706 which in turn enables gates 710 and 712. The output of gate 712 is applied over lead E to register 717 and causes all stored message elements to shift one place to the right. The message element stored in the first cell is consequently shifted into AND-gate 718, already enabled on lead D, and thence through OR-gate 719 onto line 720. At the same time the output of gate 712 is applied to stepping switch 714 over lead 729 to shift the latter one place to the right. It is evident that the reason for the additional cells in register 717, such as that marked "C+S," is for the purpose of storing additional input bits while the start signal is being generated.

Succeeding output timing pulses similarly shift the contents of register 717 to the right and onto the line until counter N has counted to N. At this time lead 737 is activated and after an interval determined by delay element 728 lead 738 also is activated. The output on lead 738 resets information flip-flop 711. The latter flip-flop upon resetting delivers no further enabling output on lead B to either AND-gates 716 or 710. Therefore, no further bits are shifted onto the line 720. Register 717, however, continues to store input bits until cell C is reached and the cycle above described repeats. Block counter 725 resets itself to normal in due course.

During the transmission of each block of N bits input bits continue to be received. From time to time then an input bit is received at the same instant as an output timing bit. Since switch 714 and register 717 are sub-

ject to control by both the input sync and output timing bits, a potential conflict exists which could cause the loss of an input bit if special provision were not made. How this is accomplished is explained below.

FIGS. 8 and 9 are timing diagrams showing the operation of the transmitting buffer shown in FIG. 7 with respect to pulses on the several leads having letter designators. The slowest and fastest permissible input rates for an illustrative embodiment are separately diagrammed. It is assumed that the buffer of FIG. 7 is designed for a system having a line rate  $R_L$  of 1000 bits per second and an input rate fluctuating between 400 and 800 bits per second. One bit is assumed to suffice for a start signal. From Equation 6 the nominal buffer capacity  $C$  is two bits and from Equation 9 the transmitted block length is five bits.

FIG. 8 shows the timing relationships for an input rate of 400 bits per second. The top line represents line timing clock bits occurring every millisecond on line 703. The second line represents message bits on line 702 and accompanying sync bits on line 701 occurring every 2½ milliseconds. In order to allow a safety factor in a practical embodiment in the event that the input rate may drop slightly below the minimum rate permitted and to handle the initial filling of the buffer it is advisable to gate the information flip-flop on the next input bit after the buffer is full. Accordingly, message bit A is stored in cell 1, bit B in cell 2, and bit C in cell 3, and the stepping switch is moved to the left on the arrival of each accompanying input sync bit. Lead A (third line) is activated on the arrival of bit C and sets the information flip-flop 711 which in turn enables gate 716 on lead B (fourth line) to pass output timing pulses to the timing flip-flop 706. The latter flip-flop stores the timing pulse for a short interval until gate 710 is fully enabled as an indication that there is not an input sync bit arriving simultaneously. Otherwise, a race condition might prevail in which both input and output bits are attempting to step switch 714 in opposite directions at the same time.

An impulse appears on lead C (fifth line), but because gate 712 is inhibited by the absence of an output on lead D (sixth line), this impulse cannot shift any bits stored in register 717 onto the line. Instead this impulse by traversing lead 727 starts block counter 725 and over lead G (line nine) starts code generator 721 also. After counter 725 has counted the first output timing pulse on lead C a delay less than an output timing interval transpires in element 722 and lead D is activated, thus enabling gates 718 and 712. The next output timing pulses therefore activate lead E (seventh line) as well as lead C to shift bits from register 717 onto line 720. While bits A, B and C are being shifted onto line 720, bits D and E are being stored in register 717.

It will be observed that four stepping switch positions are indicated on the fourth line from the bottom, but that bits are never stored in other than positions 1, 2 and 3 of the register. The stepping switch positions indicated are those reached after the last shift has occurred. In the case of bit E which arrives at the same time as an output timing pulse, either storage cell 1 or 2 is filled, depending on which of the input or output timing bits gains control of the stepping switch 714. The second line from the bottom shows what message bit is currently stored in the first or readout position of the buffer.

The cross-connections over leads 726 and 731 between timing flip-flop 706 and gate 707 on the one hand, and between gates 707 and 710 by way of inverter 709 on the other hand, assure that whichever sync bit gains control retains control and inhibits the other until its task is accomplished. This also explains the use of the input flip-flops 704, 705 and 706 to provide temporary storage of the input pulses in the event of a race condition.

After counter 725 has reached N, in this case 5, information flip-flop 711 is reset, thereby removing the output on lead B and inhibiting the output timing pulses from

the buffer. The message structure on the output line is shown on the last line of FIG. 8 for the slowest input bit rate. Each block contains five bits, including the start bit, followed by five fill-in or zero bits.

A similar timing diagram for the assumed maximum input bit rate of 800 bits per second is shown in FIG. 9. At this input rate it is seen that the message is delivered to the output line 720 in FIG. 7 continuously with no fill-in bits between blocks and with the only separation between blocks provided by the start signal bits. As in the slow input case diagrammed in FIG. 8, the buffer is initially filled to capacity by message bits A and B and the following bit C triggers the information flip-flop, thereby connecting the output timing source to the start code generator. Inasmuch as bit D arrives at the buffer before the first message bit is applied to the outgoing line, an additional storage cell is required in the buffer, making a total of four. The stepping switch is supplied with an additional step five (diagrammed on the fourth line from the bottom in FIG. 9) which, however, need not connect to a buffer storage cell because an output timing bit occurs before the next input bit to shift the switch back to the fourth position as is evident between input bits D and E, H and I, and so forth. Input bit E is seen to coincide with an output timing bit and a potential race condition again occurs. However, regardless of whether the input sync bit or the output timing bit seizes control of the stepping switch, the input bit is fed into position four of the buffer and stored there. A short delay interval (determined by element 722) after the counter has counted the end of the block, output on lead D ceases and prevents any further message bits from being delivered to the line. An output occurs on lead 737 followed by one one lead 738 to reset the information flip-flop 711. Coincident with the output on lead 738 counter 725 is reset. On the very next input bit G the buffer fills, lead A is activated and information flip-flop is again set and the next output timing bit restarts the counter 725 and also the start code generator 721. Effectively there is continuous operation of the counter and message bits are delivered to the line separated only by the start bits.

It will be understood that in actual operation of a practical system the input bit rate may vary in any manner between the slowest and fastest input bit rates and fill-in bits are normally to be expected between message blocks varying in number from five at the slowest input rate to none at the maximum rate. The fourth storage cell for the buffer used at the highest bit rate is only in the nature of reserve capacity and would be unnecessary if the input bit rate were allowed merely to approach 800 bits per second but not actually to reach it. If additional bits are used in the start signal, appropriate additional storage cells are to be provided in the buffer.

At the receiver two basic arrangements are possible depending on whether the message is to be accepted at a clock rate determined by the receiver or at the line rate. Two practical embodiments are diagrammed in FIGS. 10 and 12. The embodiment of FIG. 10 is very similar to that shown in FIG. 7 for the transmitting buffer with the addition of a start code detector, while that shown in FIG. 12 eliminates the receiving buffer.

The receiving buffer of FIG. 10 includes logic elements similar to those found in the transmitting buffer shown in FIG. 7, and related elements are designated by the same tens and units digits. However, the input synchronization on line 1001 is the fixed transmitting rate  $R_L$  and the data-out timing on line 1003 is determined at the receiver and is assumed to be below the line rate but preferably about the average input bit rate. The recovered sync rate on line 1001 is derived from the composite message in any suitable manner, such as by the use of a flywheel oscillator or tuning fork, as disclosed, for example, in a copending United States application Serial No. 692,174, filed on October 24, 1957, by A. D. Perry,

## 11

Jr., now Patent No. 2,957,045, issued October 18, 1960. Input lines 1001 and 1002 are connected by lines 1018 and 1019 to start code detector 1021, which recognizes the first coincidence of the start signal bit and a recovered sync bit at the beginning of a message block. Upon recognition of a start signal the start code detector 1021 furnishes a momentary output to information flip-flop 1011 at the input marked "S." Immediately there appears an output on lead B joining flip-flop 1011 and AND-gates 1012 and 1016 to enable the latter gates. On the occurrence of the following sync pulse, sync flip-flop 1004 is set to produce an output on its output lead designated "1." If the accompanying message bit on line 1002 is a binary one, flip-flop 1005 is set and produces an output on the lead marked "1." The outputs of flip-flops 1004 and 1005 control AND-gates 1007 and 1008, respectively. Gate 1007 additionally has a normally enabling input from timing flip-flop 1006 which becomes inhibiting only when a data-out timing pulse is simultaneously present to avoid a conflict between sync and timing pulses over control of register 1017.

An output from gate 1007 on lead 1031 enables gate 1008 to allow the message bit to be stored in register 1017 over lead 1032. This same output is available on lead 1027 to reset both sync and data flip-flops, and further on lead C to move stepping switch 1014 to the left after a short delay through element 1013 to lead D. The AND-gate of the group designated 1015 which is enabled at a given time is determined by the position of the stepping switch 1014 as in the transmitting buffer. The output on lead C further operates the block counter 1025. This counter need only count the actual intelligence bits in the message block in contrast to the counter used in the transmitter inasmuch as the start signal is effectively counted separately in detector 1021. In any event, as the end of the transmitted block is counted, counter 1025 emits a pulse to delay element 1028 which becomes the output on lead I. The delayed pulse on lead I resets information flip-flop 1011 and start code detector 1021. The enabling signal is then removed from lead B and gates 1012 and 1016 are blocked until the next start signal is detected.

In the meantime the data-out timing signals on lead 1003 have been arriving continuously, each time setting timing flip-flop 1006. Timing flip-flop 1006 enables gate 1010 in the set state and provided there is no output from gate 1007 through inverter 1009, an output pulse appears on lead E. The cross-connection between gates 1007 and 1010 through inverter 1009 is to prevent the occurrence of a race condition as previously explained. The lead E output causes a shift of the contents of register 1017 to the right, thereby delivering a message bit to data-out lead 1020 for each pulse on lead E. If desired, buffer full and empty signals can be arranged in case of operation outside the normal bit rate limits. These signals may be made available on leads 1033 and 1034 as shown.

FIG. 11 is a timing diagram showing the operation of the receiving buffer of FIG. 10 when receiving the message furnished to the transmitting buffer of FIG. 7 at a rate of 400 bits per second. The messages in the timing diagrams of FIGS. 8 and 11 are the same. FIG. 11 shows how that message is reconstructed at the receiving terminal having the embodiment shown in FIG. 10.

The recovered sync on the first line is at the line rate of 1000 bits per second. The message on line 2 occurs in five-bit blocks including a prefatory start bit and followed by five fill-in bits. The third line shows the occurrence of an output from the start detector whenever the S bit appears. This S bit in turn sets information flip-flop 1011 to give an output on lead B (fourth line) until counter 1025 counts the intelligence bits and resets the information flip-flop. As long as there is an output on lead B pulses occur on lead C in synchronism with the recovered sync pulses. As soon as switch 1014 under

## 12

the control of pulses on lead D steps to position 2 the signal is removed from lead A indicating that the buffer is filling. Counter 1025 is operating as shown on line ten in accordance with pulses on lead C. Data-out timing pulses are occurring continuously on lead E, subject only to a slight delay in the event a recovered sync pulse on lead C occurs at nearly the same time. Pulses on lead E shift the contents of the register onto line 1020 and also step switch 1014 to the right.

The successive positions of switch 1014 are shown in line eleven. A shift to the left occurs with every pulse on lead D and to the right with every pulse on lead E. Positions where one digit occurs over another, as at the occurrence of the first pulse on lead D simultaneously with a customer timing pulse on lead E, indicate that the switch is stepped in one direction momentarily and then rapidly shifted back to the same position.

On the second line from the bottom is shown the reconstructed message, each bit occurring with a pulse on lead E. No start bits are found in the final message because they were absorbed in detector 1021. The customer need not be aware of the form his message took while passing over the synchronous transmission system. If the customer somehow has knowledge of the form in which the message was delivered at the transmitter, he can vary the data-out timing as necessary to reproduce the original message according to a fluctuating bit rate in the embodiment of FIG. 10.

A simplified receiver terminal is shown in FIG. 12 for use where a receiving customer is willing to accept the message encoded into blocks. No buffer is required, but only a start code detector and a block counter. Recovered sync bits on line 1201 and data bits on line 1202 are applied to AND-gates 1208 and 1209, respectively. These gates are blocked unless information flip-flop 1206 has been set. A start bit occurring on line 1203 is brought to start code detector 1205 with a sync bit on line 1203. Detector 1205 recognizes the start bit and gives an output through delay element 1207 to the set input of flip-flop 1206. Gates 1208 and 1209 are thereby enabled. Input bits are admitted to lines 1210 and 1211. Counter 1213 counts sync bits on line 1210 by way of line 1212 until the count  $N-S$  is reached. The S bit is absorbed in start code detector 1205. Counter 1213 on the last count gives an output through delay element 1214 which resets detector 1205 and flip-flop 1206. Detector 1205 stands by until another start bit is recognized. In this case the recovered message on data-out line 1211 is in block form as shown in FIG. 8, for example, on the last line, with the start bits deleted, however.

The preceding description has demonstrated by means of particular illustrative embodiments that by the use of the buffer storage method of this invention asynchronous data sources can be continuously connected to synchronous transmission systems and that the original message can, if desired, be recovered in asynchronous form. However, the principles of this invention are not to be deemed limited to the particular embodiments described above nor by the bit rates specified in the examples.

What is claimed is:

1. In a data transmission system an arrangement for matching an asynchronous digital input signal to a transmitter operating at a synchronous rate higher than the average asynchronous rate comprising a digital data source supplying information bits and synchronizing bits at said asynchronous rate, an output sink adapted to receive information bits only at said synchronous rate, a synchronous clock source independent of said data source, a register of fixed capacity for temporary storage of said information bits interposed between said source and said sink, means for continuously shifting into said register under the control of said input synchronizing bits said information bits at said asynchronous rate, means for generating a start-of-block signal whenever the capacity of said register is reached, means coincident with said



13

14

start signal for connecting said clock source to said register and means for counting a predetermined block of information bits having more elements than the fixed capacity of said register to be shifted out of said register at said synchronous rate and for disconnecting said clock source from said register after said predetermined block has been removed therefrom.

2. In a system for the transmission of digital data at a fixed synchronous bit rate, a source of binary digital data signal bits continuously emitted at a fluctuating time rate, a transmission line adapted to transmit data signals at a fixed synchronous time rate, and means interposed between said source and said line for translating continuous signals from said source into uniform length blocks of signals for transmission at said synchronous rate comprising a shifting register having a fixed number of storage cells less than the number of elements in said uniform block length, a switching network for gaining access to individual ones of the storage cells in said register, connecting means between said source and said network for advancing the operation of said network one step in one direction for each signal bit stored in said register, a synchronizing clock source determining said synchronous transmission rate, means responsive to the filling of the fixed number of storage cells in said register for connecting said clock source to said network and register, said network stepping in the opposite direction for each of said clock pulses and the contents of the cells in said register being shifted onto said line one at a time for each clock pulse, a start-of-block code signal generator, a first output from said generator to said line for applying a start code signal thereto, a second output from said generator for inhibiting the connection of said clock source to said register until the start code signal has been transmitted over said line, and a counter connected to said clock source for sensing the number of signal bits applied to said line and for disconnecting said clock source from said register after the transmission of a predetermined block of signal bits from said register over said line.

3. A synchronous transmission system for binary digital data message bits comprising a transmitter, a receiver, a line joining said transmitter and receiver, means at said transmitter for temporarily storing a fixed number of message bits generated at a fluctuating rate, means for generating a start signal whenever said fixed number of message bits has been stored, a source of synchronizing bits, means for connecting said synchronizing source to said storing means upon termination of said start signal, a counter for controlling the connection of said clock source to said storing means until a predetermined number of message bits exceeding said fixed number has been fed to said line; and at said receiver further means for temporarily storing message bits received over said line, means for recognizing the occurrence of said start signal, means operative in response to the recognition of said start signal for connecting said further storing

means to said line, a source for generating timing bits at a rate slower than said synchronizing rate, and means connecting said last-mentioned source to said further storing means for emptying said further storing means.

4. In combination, a source of binary data signals occurring at a bit rate fluctuating between a fixed upper and lower bound, a sink for binary data signals adapted to accept such signals at a single fixed bit rate at least as high as the upper bound of said fluctuating rate, a fixed capacity buffering means interconnecting said source and sink, means for continuously delivering signals from said source to said buffering means, and means for transferring a predetermined fixed number of said signals from said buffering means to said sink at said fixed bit rate upon the filling of said buffering means to capacity, the capacity of said buffering means being substantially less than the predetermined number of signals delivered to said sink at one time.

5. The combination according to claim 4 in which the capacity of said buffering means equals the product of said predetermined fixed number of signals withdrawn from said buffering means at one time and the ratio of the difference between the upper and lower bounds of said fluctuating bit rate and said fixed bit rate.

6. In a system for transmitting digital data signals at a fixed synchronous bit rate, apparatus for normalizing a source of digital data signals having a bit rate which fluctuates between preassigned upper and lower limits below said fixed rate comprising means for continuously storing said input bits at said fluctuating rate, means for generating a transmission start-of-block output signal whenever a predetermined number of input bits has been stored, means for transmitting said start signal and a message block of preassigned length exceeding said predetermined number of input bits at said fixed bit rate, and further means for transmitting fill-in bits at said fixed bit rate following said message block until a further start signal is generated.

7. In a system for transmitting digital data signals at a synchronous bit rate, apparatus for buffering a source of digital data signals having a bit rate fluctuating between preassigned upper and lower bounds below said synchronous rate comprising means for continuously storing said input bits at said fluctuating rate and means for transmitting a block of data signals of predetermined length at said synchronous rate after a certain number of input bits has been stored, the length of said transmitted block being related to said certain number by the ratio between said synchronous bit rate and the difference in the upper and lower bounds of said fluctuating bit rate.

#### References Cited in the file of this patent

##### UNITED STATES PATENTS

|           |            |                |
|-----------|------------|----------------|
| 2,907,004 | Li Chien   | Sept. 29, 1959 |
| 2,913,705 | Cox et al. | Nov. 17, 1959  |
| 2,991,452 | Welsh      | July 4, 1961   |