SYSTEM CENTURY®
DIGITAL SWITCHING TECHNIQUES

CONTENTS

ILLUSTRATIONS

1. INTRODUCTION

1.01 This section contains basic information related to digital switching techniques used in SYSTEM CENTURY Digital Central Office (DCO) and Digital Branch Exchange (DBX) equipment. Discussed in this section are the analog-to-digital encoding process, the pulse code modulation process, the time-division multiplexing process, and the digital-to-analog decoding process.

1.02 This section is reissued to provide information on the digital switching network used in the SYSTEM CENTURY Digital Branch Exchange (DBX). Significant changes and additions are indicated by bars in the margins.

2. RELATED INFORMATION

2.01 General.

A Sales and Instructional Literature Index of the latest available publications can be obtained from your Stromberg-Carlson sales representative or from Publications Services, Stromberg-Carlson Corporation, P.O. Box 1698, Sanford, Florida 32771.

2.02 Related Publications.

The following publications are recommended for use with this document:

01-025-05	Glossary of Digital Switching Terms and Abbreviations
23-001-02	SYSTEM CENTURY DCO, General Description
53-001-02	SYSTEM CENTURY DBX-1200, General Description

3. DIGITAL SWITCHING TECHNIQUES

3.01 SYSTEM CENTURY Digital Switching Systems.

The Stromberg-Carlson family of digital telephone switching systems features stored-program control (software control) and uses time-division multiplexing (TDM) and pulse code modulation (PCM). The stored-program concept includes a special telephony preprocessor with an associated programable memory having sufficient capacity to maintain the transient call status of each port (line/trunk) in the digital switch. Each port, therefore, has dedicated, real-time, special-purpose preprocessor functions that do not interrupt the sequential call-handling tasks of the call processor. With this control concept, switches can be updated as new features/services are identified and offered. Stromberg-Carlson's unique software package controls system operations. This package includes specific programs for the handling of call operations and software programs for the maintenance (routine/automatic) and administrative operations (features) of this digital switching system.

3.02 Basic Techniques, Line-to-Line Call.

In digital switching systems developed by Stromberg-Carlson, the speech or intelligence from one user is first converted from analog to digital, using PCM. Up to 30 ports are multiplexed onto 1 channel using time-interleaving, and then this PCM information is forwarded to a timeslot interchange (TSI). Elements (memory, etc.) within the TSI result in time-division multiplexing of the speech paths and, after timeslot assignment, are converted back into analog information for a second user. See figure 1.

3.03 Analog Versus Digital Matrix.

The use of TDM results in a substantial reduction in the size of the switching matrix. In an analog matrix, one path carries only one conversation. In the DCO, 1 path carries 128 independent conversations simultaneously. One path in the DBX can carry up to 192 conversations. The conversation sampling rate is so fast that the subscribers are unaware of the small (microsecond) separations between bits of conversational information.

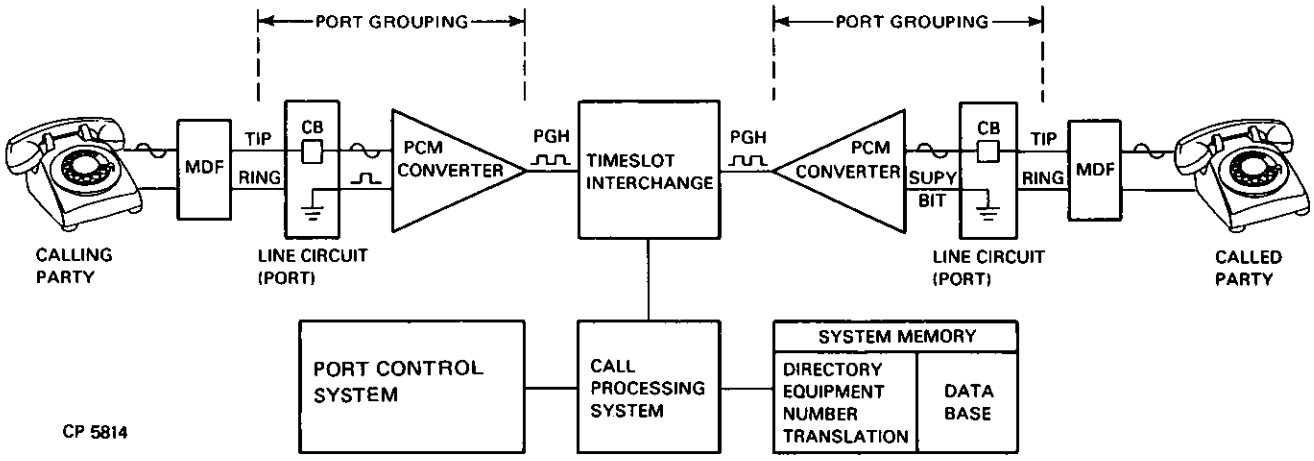
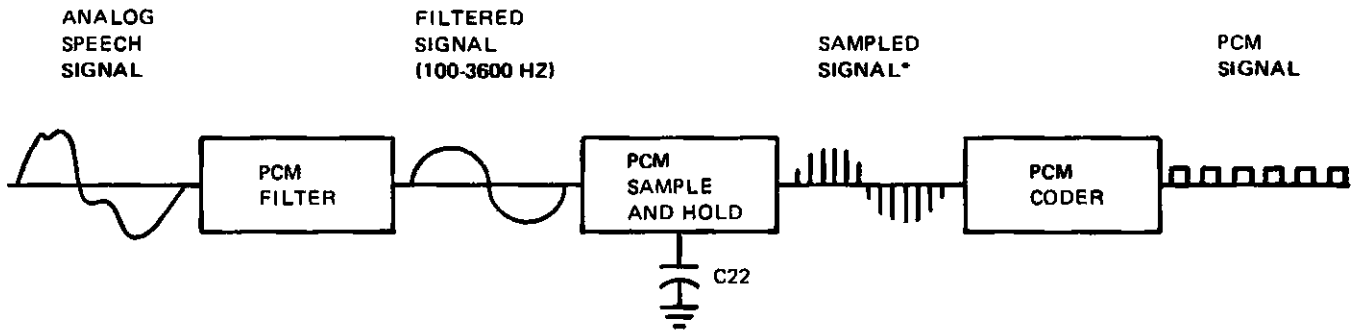


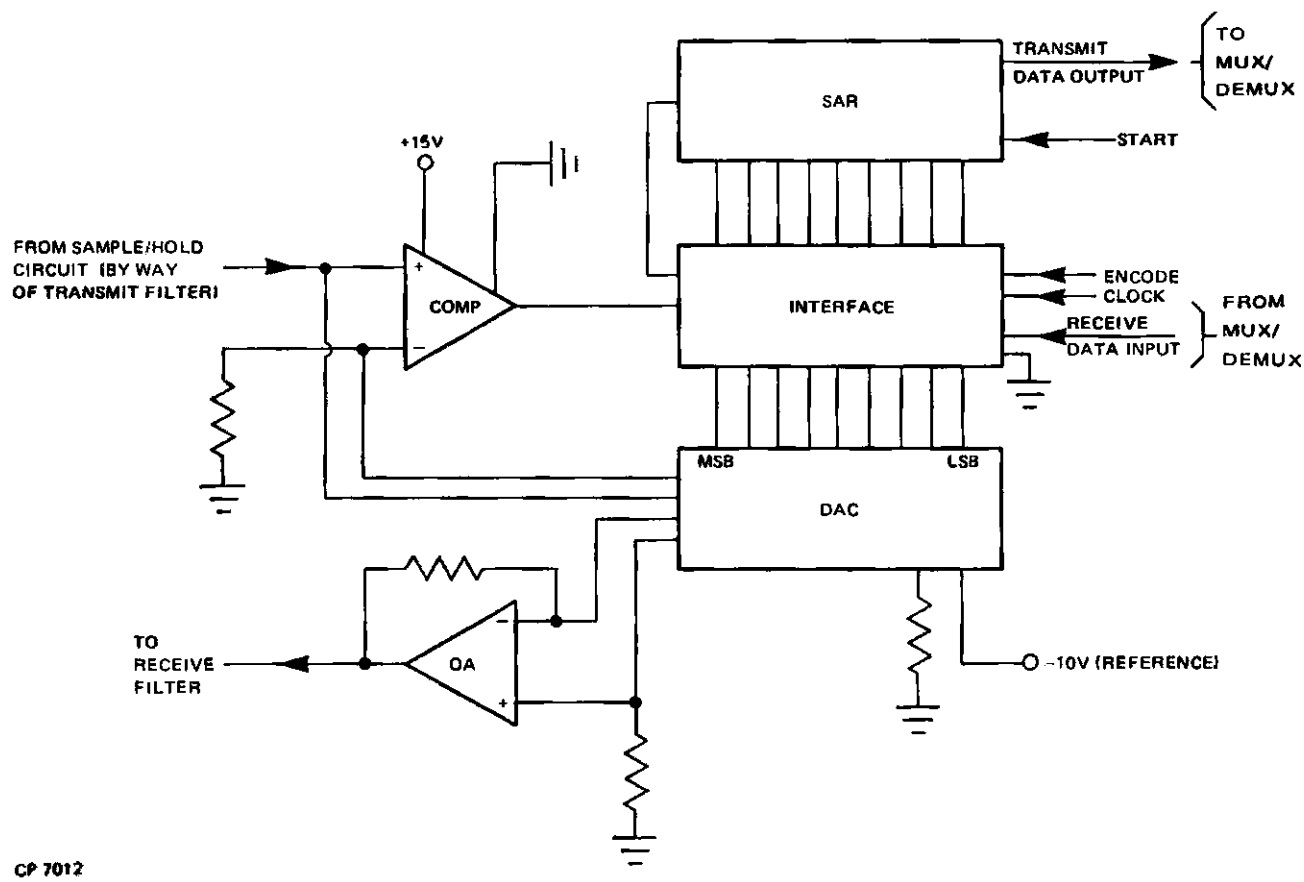
Figure 1. Line-to-Line Call



*Sampled for 54.7 microseconds. Held in C22 for 70.3 microseconds.
54.7 + 70.3 equals 125 microseconds or 8000 times/second.

CP 5815

Figure 2. Analog-to-Digital Basic System



CP 7012

Figure 3. CODEC

4. ANALOG-TO-DIGITAL CONVERSION (ENCODING)

4.01 Introduction.

The analog-to-digital (A/D) system starts with an analog speech signal present on one of the input/output (I/O) ports. This signal is passed through a transmit filter where all frequencies above 3600 Hz and below 100 Hz are filtered out. This filtered analog signal is then sampled for 54.7 microseconds and held for 70.3 microseconds. From a sample/hold circuit, this signal is then processed by a coder. The CODEC (coder-decoder) encodes each signal into an 8-bit digital word. This A/D basic system is shown in figure 2. The signal is now discrete in both amplitude and time.

4.02 CODEC.

The CODEC is an integrated circuit that does the A/D and D/A conversions. This integrated circuit is an assembly of five subcomponents. These subcomponents are the comparator (COMP), successive approximation register (SAR), interface, digital-to-analog converter (DAC), and the operational amplifier (OA). The CODEC is shown in figure 3. The A/D conversion is done with the DAC, COMP, and SAR, using the successive approximation (SA) technique; this is described later. The DAC is part of a feedback loop where the DAC's output is compared with the analog input. When the interface makes these equal, the correct digital encoding is achieved. The D/A conversion is done with the digital-to-analog converter. The DAC is used both for A/D and D/A operations. For a D/A conversion, the signal from the DAC is fed into the operational amplifier. The signal out of the OA is in the (50-percent duty cycle) pulse amplitude modulation (PAM) form and goes to the receive filter. The interface sends appropriate data to either the SAR or the DAC, according to the conversion process selected. The CODEC gets its synchronization and clock signals from the system timing circuits to control A/D and D/A conversions. The generation of these signals is shown in figure 4. Each CODEC in the port input/output system has an associated transmit filter, receive filter, and a sample-and-hold (S/H) circuit. The S/H circuit is used for the A/D process only. Both receive-data-in (at the interface) and transmit-data-out (at the SAR) are in serial, nonreturn-to-zero form. This means that the pulses do not return to some reference level between bits.

4.03 Successive Approximation Technique.

The actual A/D conversion is done by the CODEC circuit (subcomponents DAC, COMP, and SAR), using the SA technique. This technique operates by comparing the analog input (COMP in figure 3) to a series of bit trials. The first trial compares the input with the value of the most significant bit (MSB). Figure 5 shows the progression of trials for a 3-bit converter. If the input is greater than the MSB value, the MSB is retained and the converter tries the next MSB. If the input had been less than the MSB, the interface logic normally turns the MSB off before going on to the next MSB. This branching continues until each successively smaller bit has been tried, with the entire process taking "n" trials.

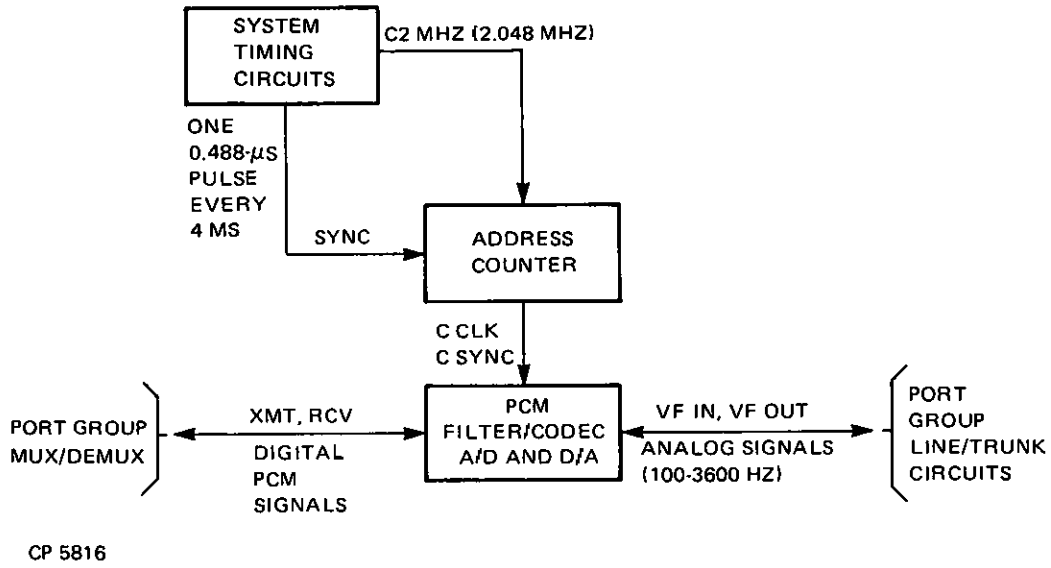


Figure 4. CODEC's Signals

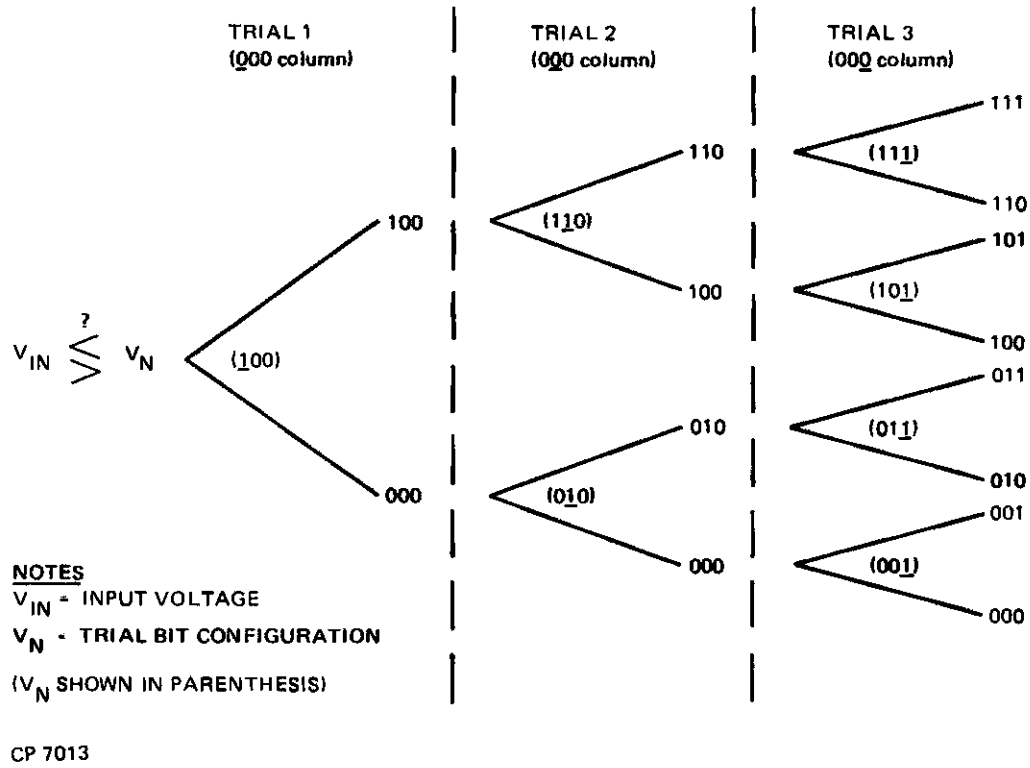


Figure 5. Three-Bit Successive Approximation A/D Conversion

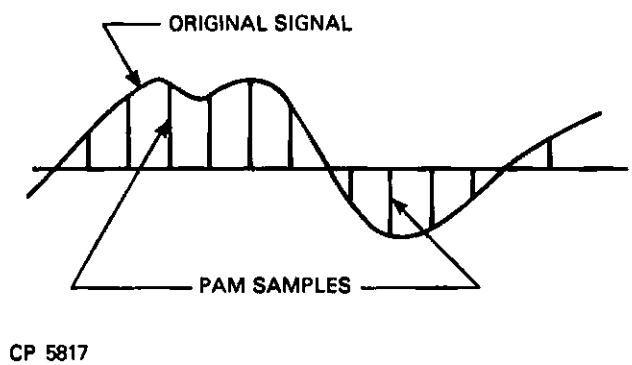


Figure 6. Sampling Process

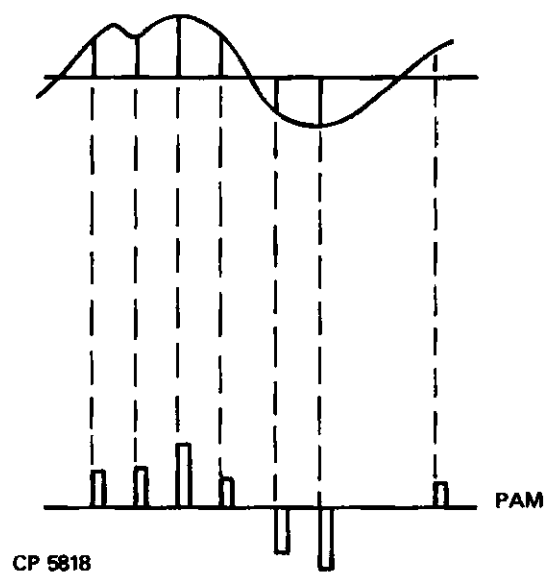


Figure 7. Expanded Sampling Process

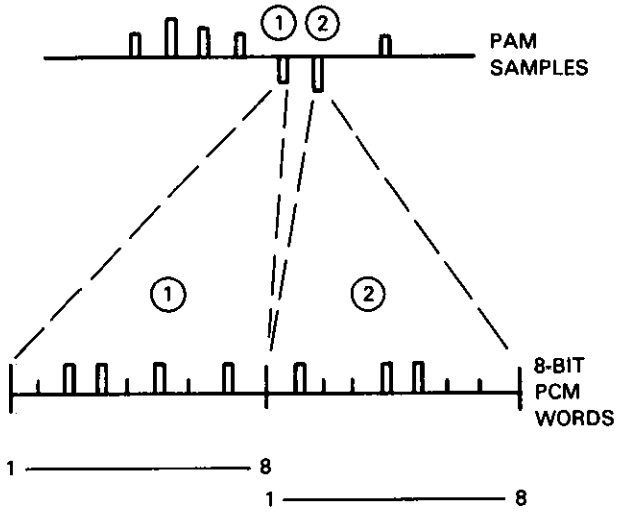


Figure 8. PCM Word Creation

4.04 Pulse Code Modulation Process, Background Information.

a. The PAM signal at the coder input consists of sequential amplitude samples of the original analog port signal. This sampling process is shown in figure 6.

b. Taking several of these PAM samples and expanding (see figure 7), since each PAM pulse has an absolute pulse width, the CODEC's input is defined more precisely.

c. The CODEC (PCM coder) takes this input and creates an 8-bit PCM word for each sample. See figure 8. The resulting PCM word is a code, represented by a series of equal amplitude 1s and 0s, that corresponds to its associated PAM sample. In the preceding figure, example 1 reads 00110101 and example 2 is 01001100.

d. A simplified coder scheme for deriving these codes is shown in figure 9 (3-bit scheme). A coder input (V_{IN}) anywhere between X_2 and X_3 results in the code word 010. A decoder input of 010 produces the level Y_2. The overall effect of the CODEC is that of a quantizing staircase transfer function.

e. One way for coders to represent sample amplitudes is by a symmetrical (or sign-magnitude), 8-bit binary code. In this system, the first bit represents the sample polarity (0 = positive, 1 = negative), and the remaining 7 bits encode sample magnitudes from 0 to 127. Thus, there is a total of 255 levels (0 is represented by 2 codes: ±) ranging from −127 (11111111) to +127 (01111111). The symmetrical codes achieve the desirable step-size gradation, shown in figure 10, in a piece-wise linear manner. The process that yields this staircase is called "companding" or nonuniform coding. This process results in a nearly constant signal-to-noise ratio over a 40-dB range. This 40 dB is significant since it represents talkers (voice) range in volume. Without companding, the weaker signals would have a very poor signal-to-noise ratio. With the application of a PCM coding law to the preceding "companding function" figure, real numbers are assigned to each decision level. Applying the "mu" Coding Law, the coding magnitude range is divided into eight segments. There are 16 levels spaced within each segment. From one segment to the next higher, the level spacing (step size) increases by a factor of 2.

f. In the 8-bit code word representing any sample (see figure 11), the first bit gives the sign, the next 3 bits describe which of the 8 segments contain the sample, and the last 4 bits specify the nearest of the 16 possible levels within the segment. The "mu" Law code is digitally linearizable (that is, the code is readily convertible from compressed to linearized form) using simple digital operations. The compressed form is ordinarily transmitted; the linearized form is the code that results if the first segment were extended with the same step sizes to cover the entire input range. A linearized code is desirable if digital processing of the signal is required. For example, two digitized speech signals can be added together (as in a conference call) by converting each to linearized form, adding them by ordinary binary arithmetic, and converting the sum back to compressed form for standard, nonuniform decoding. This can be done entirely in the digital domain much easier than converting to analog, summing, and then converting back to digital. These foregoing concepts are applied to figure 12.

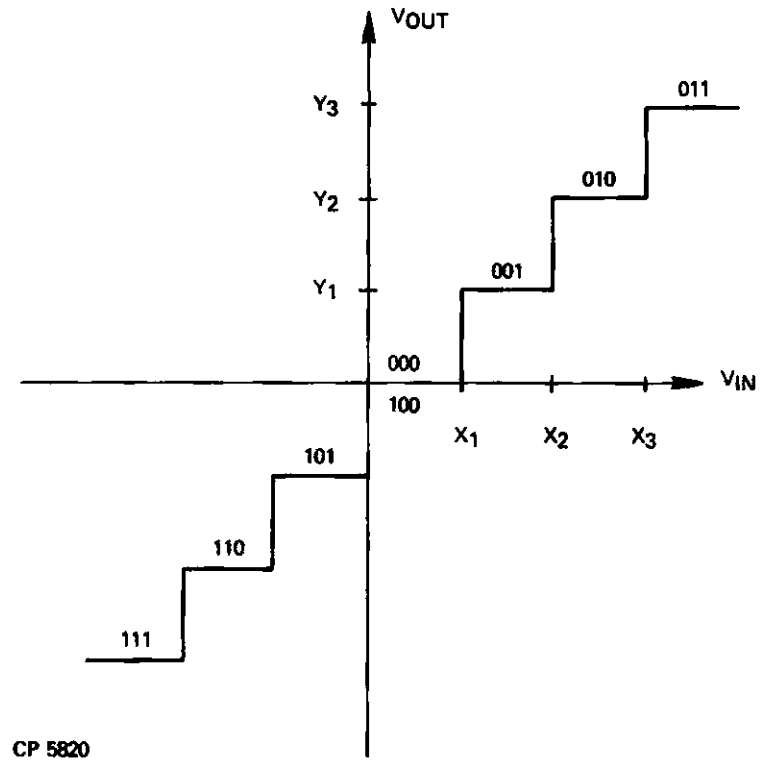


Figure 9. Simplified 3-Bit Coder Scheme

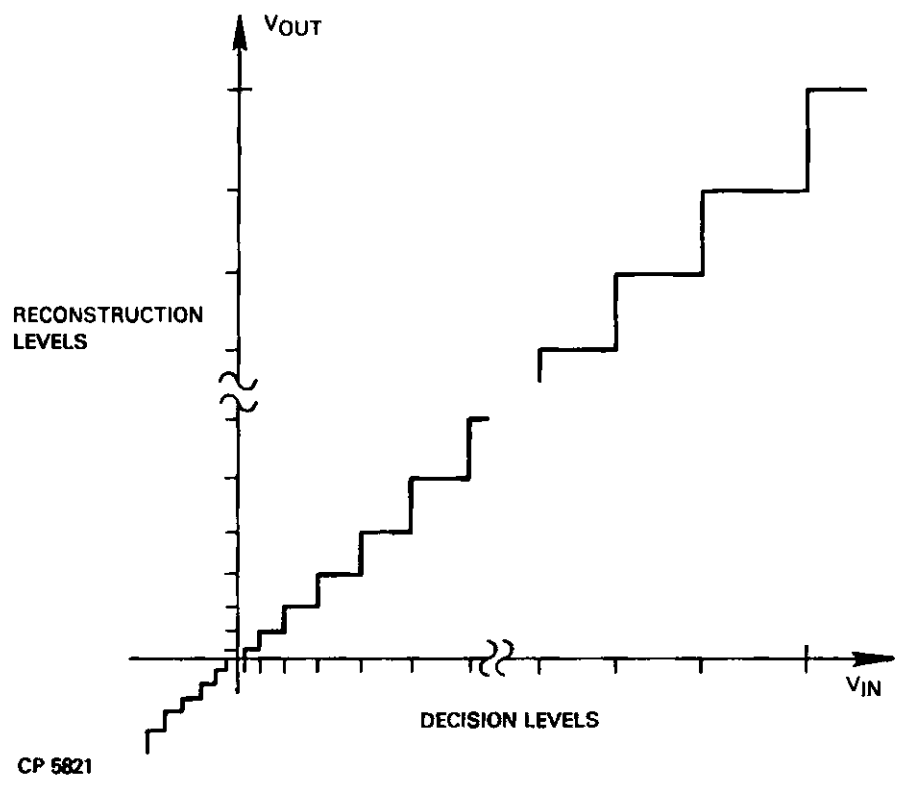


Figure 10. Companding Function

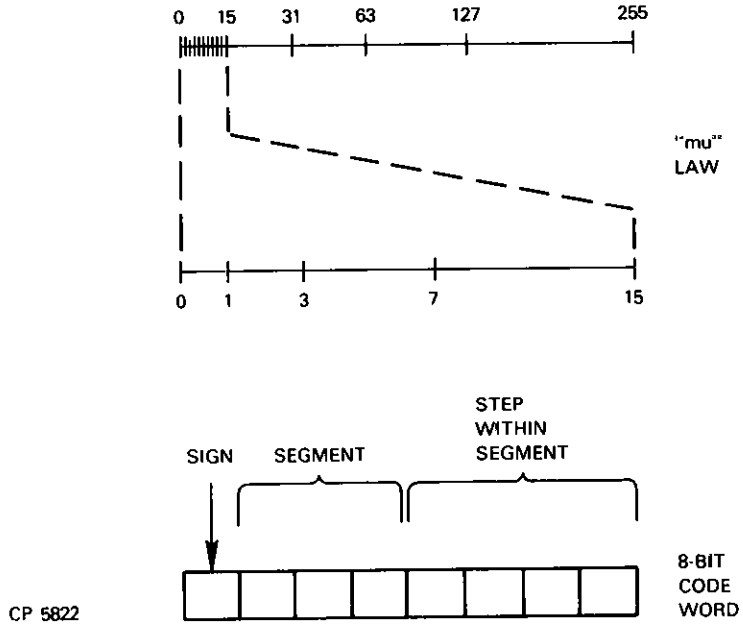


Figure 11. Companded PCM "mu" Law

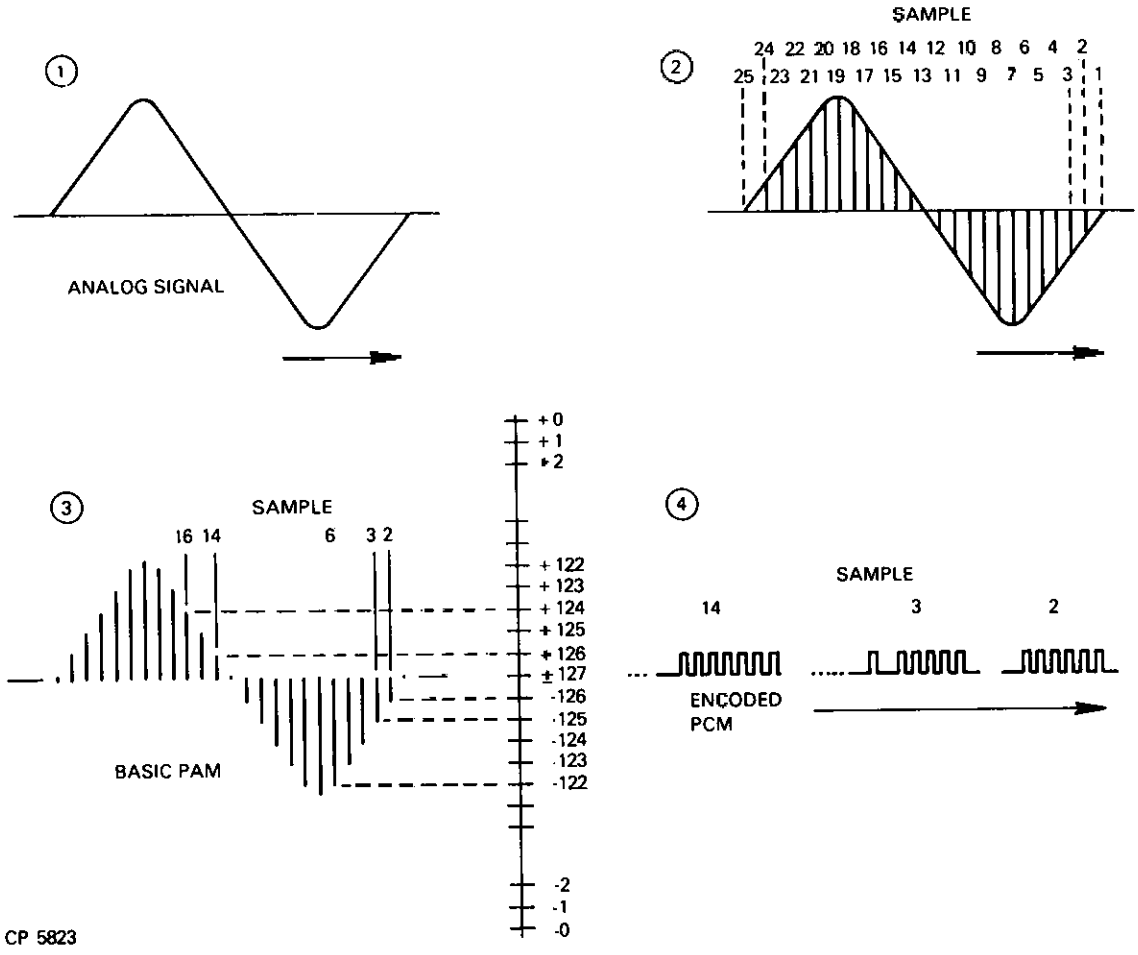


Figure 12. Basic PCM Concept

5. TIME-DIVISION MULTIPLEXING

5.01 Introduction.

The operation of an elementary TDM system is dependent on the ability to connect two lines together for very brief periods at regular intervals. In a system that uses more than one pair of input/output terminals, a method of controlling called and calling lines is essential. In this system, called/calling line information is stored in associated memory devices according to equipment numbers. When a match of these numbers is found, a highway (path) is then allocated for voice communications. These paths are called cross-office highways (XOHs). The timeslot interchange (TSI) memory devices are random access memories (RAMs) for nonsequential, rapid addressing of stored equipment numbers and XOH information. These memory addresses correspond to a slot in time so that many paths can be active simultaneously.

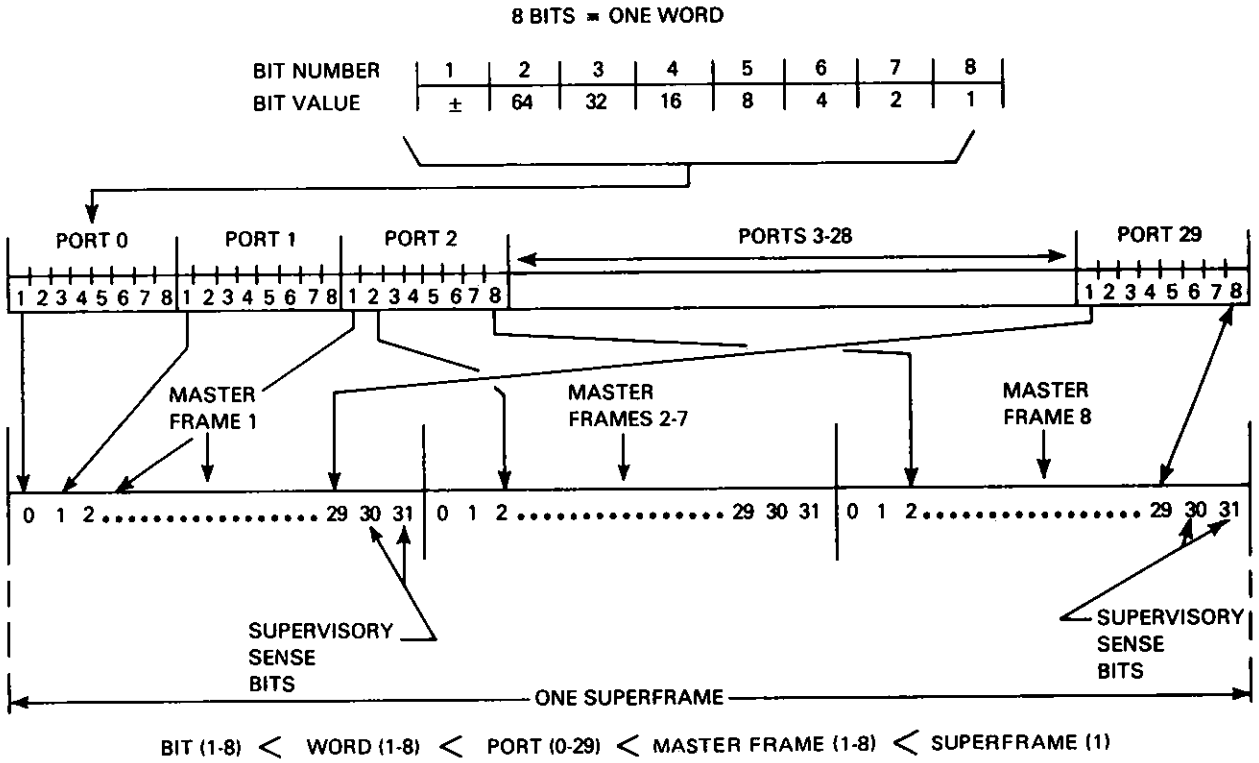
5.02 SYSTEM CENTURY TDM.

The basic PCM encoding and decoding processes are similar in both the SYSTEM CENTURY Digital Central Office (DCO) and Digital Branch Exchange (DBX). However, because the two systems have substantially different applications, each uses a different method for multiplexing data and switching digital signals. Therefore, the digital switching networks in the DCO and DBX are described separately in the paragraphs that follow.

6. DCO DIGITAL SWITCHING NETWORK

6.01 Mutltiplexing Data.

In the DCO, the data for a group of 30 ports is multiplexed into a serial bit stream before entering the TSI. The 8-bit coded words from all 30 ports (0 through 29) are combined into a frame of port words. A frame is formed by extracting the same bit from each 8-bit port word. All the 1 bits from the 30 ports combine as frame 1. The 2 bits are used to derive frame 2, etc., up to 8 bits, which combine as frame 8. These eight master frames constitute a superframe. (See figure 13.) After the port group control adds the supervisory sense bits (bits 30, 31) to each master frame, the serial bit stream (formatted as a superframe) of 2.048 Mb/s is sent to the port group highway (PGH). This interfacing highway between the port I/O and the TSI hardware subsections represents multiplexed audio and control information (TDM) from up to 30 ports. Two hundred and forty



Notes:
1 Superframe = 32 bits from 8 master frames or 256 bits
Master frame rate = 8 kHz
Bit rate = 2.048 Mbits/sec

CP 5824

Figure 13. DCO Port Group Highway Format

timeslots are reserved for the ports, and 16 timeslots are reserved for supervisory sensing on each superframe-formatted PGH. The PGH format is shown in figure 13.

•6.02 Timeslot Interchange.

The timeslot interchange subsection of the DCO system provides for the switching of 64 kb/s (from the CODEC's A/D converter) from port to port (figure 14). The TSI interfaces with the port input/output subsystem and is directed by the common control system controller. This controller, in turn, is controlled by the call processor. All paths are established using cross-office timeslots on cross-office highways. Ports are connected to the TSI by way of port group highways. At the I/O input, each interchange handles up to 8 external port groups (30 ports each), plus the supervisory information related to these groups. The MUX/DEMUX multiplexes the data from these 240 ports into a single bit stream of 128 0.122-μs timeslots. This 128 multiplied by 0.122 μs is called a time frame (15.62-μs frame). The number of TSIs can be expanded, with up to 32 TSIs each serving a XOH. The timeslot generator (inputs to the send and receive buffers) gets its source 8.192-MHz clock from the local clock distribution (derived from the master clock). The send buffer gets inputs from the MUX, the TS generator, and the send store, and outputs as a XOH. The receive buffer gets inputs from the TS generator, the XOH selector, and the receive store, and outputs by way of the DEMUX as call data. All three stores are controlled by the common controller. These stores are address buffers and are hardware implemented with RAMs. These send, receive, and XOH devices store calling equipment number, called equipment number, and XOH timeslots, respectively.

6.03 Principles of TSI Data Switching.

Each TSI is functionally divided into a send and receive subsection. The send logic receives input (serially multiplexed voice and supervisory data from up to 240 ports) signals from up to 8 PGHs and generates this data on a single XOH that goes to the receive logic. Under program control, this logic selects data from these XOHs and switches this data to the appropriate outgoing PGH. The XOH serial data format, outpulsing at the send logic, consists of recurring 16-μs time frames. Each frame is divided into 128-bit positions called timeslots. With this 128-position format, 128 out of 240 ports can be served by the send logic at any one time. To establish communication paths, the send logic of each TSI multiplexes and stores the bits from these 240 ports, then, according to common controller directives, the send logic assigns the individual port bits to a particular timeslot (up to 128 slots). Furthermore, during each timeslot, the receive logic is selecting (by means of common controller directives) an XOH input and storing the bit present on that XOH during that timeslot. The receive logic then assigns the stored bits to a PGH. The data bits outputted from the receive logic are demultiplexed onto eight PGHs and ultimately to one port associated with one of the eight PGHs. Therefore, the communication call-path is established by assigning the same XOH TS to the calling port's send logic and to the called port's receive logic.

6.04 Details of TSI Send and Receive Sections.

a. The send section contains logic that indirectly assigns timeslots to calling ports. This logic comprises the following three elements: two RAM data buffers called send buffers; a RAM address buffer called the send store; and a recycling counter called the send timeslot generator (TSG). These elements, along with the receive section, are shown in figure 14. During each 15.62-μs time frame, port data bits are stored in one of the send buffers, while data bits stored in the preceding time frame are read from the other buffer and placed on the XOH. Since the bit location addresses in the send buffers have the same binary form as the equipment numbers (ENs) of the ports, each port has a dedicated buffer location where corresponding bits are written. During the write operation, these locations are sequentially addressed by the TSG outputs. The TSG cycles through 128 counts every 15.62 μs. These outputs simultaneously address both halves of the send buffer. While providing these write addresses, the TSG also provides write and read addresses that correspond to send-store XOH timeslots. The send store contains ENs used to address the send buffer port data bits during a read operation. The common controller loads the ENs into the send store in a sequence that requires addressing each data bit during the appropriate timeslot. As the TSG cycles, these ENs are applied as address inputs to the send buffer. The specified (by each EN) data bit location in the buffer is read out and placed on all TSI XOHs. Even though all bits from 240 ports are written into the send buffer during a single time frame, only the information associated with 128 ports is read out and placed on the XOH. The 129th subscriber waits until the subscriber load is 127 or less.

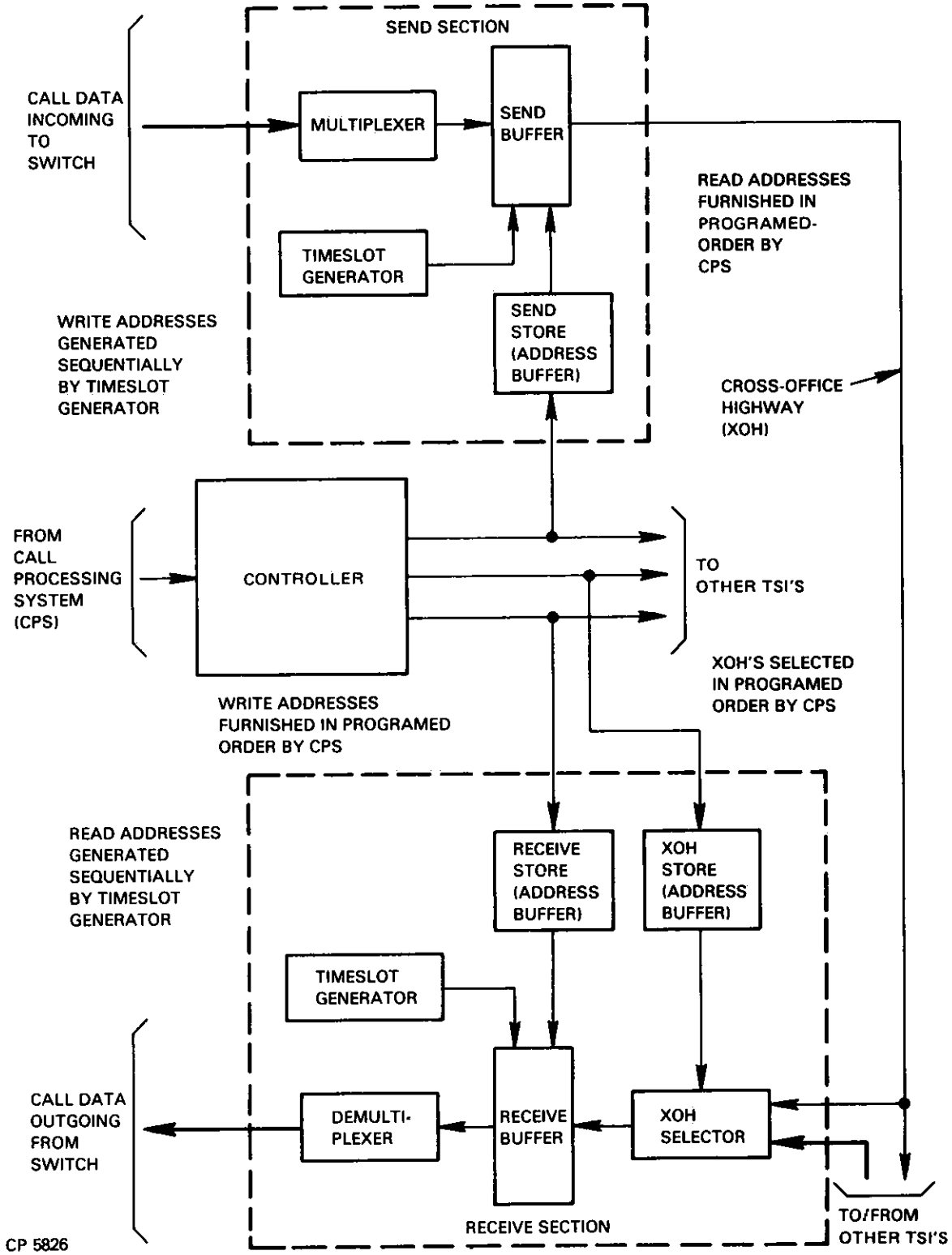


Figure 14. DCO Timeslot Interchange

b. The receive section contains logic that indirectly assigns timeslots to the called ports. This logic is comprised of the following five elements: two RAM data buffers called receive buffers; a RAM address buffer called the receive store; an XOH selector; a RAM address buffer called the XOH store; and a recycling counter called a timeslot generator. The receive TSG generates sequential read/write addresses to the receive and XOH stores. These binary addresses correspond to XOH timeslots. During initialization and, subsequently, at the start of each call, the common controller uses these addresses to load ENs into the receive store and XOH codes into the XOH store. As the TSG cycles, these ENs and codes are read from the stores and used to address receive-buffer locations and to specify to the XOH selector which XOH receives a bit. The bit, present on the selected XOH during the corresponding counter output timeslot, is gated through the selector to either receive buffer 1 or 2. During each timeslot, the TSG output addresses a receive-store location. The EN read from that location is used to address the receive-buffer location where the XOH data bit is written. During each time frame, data bits are written to the receive buffers while, from the other receive buffer, preceding time frame data bits are read and applied to a demultiplexer. At the end of each time frame, these read/write operations are reversed between the two receive buffers. During a read operation, the receive TSG outputs are interpreted by the receive buffer as ENs. These ENs, generated sequentially and applied to the address inputs of both halves of the buffer, read a single bit from each half. These bits are then stored in flip-flops while logical zeroes are written back into the original data bit locations. After this, the stored bits are strobed into a two-to-eight (2:8) line demultiplexer where they are assigned to PGHs. In the port system, these bits are further demultiplexed up to a maximum of 240 port circuits.

7. DBX DIGITAL SWITCHING NETWORK

7.01 Multiplexing Data.

a. In the DBX, each CODEC does encoding for three separate ports by sampling each of the associated ports 8000 times per second and formatting each of these samples into an 8-bit PCM word. The CODECS operate at a 384-kilobits-per-second (kb/s) rate and output one bit every 2.6 microseconds. To output one 8-bit word requires 20.8 microseconds. The outputs from two CODECS are interleaved on a single bus by using an alternating process that allows one CODEC to output while the other CODEC is turned off. To output six 8-bit words, three from each CODEC, required approximately 125 microseconds. The outputs from 10 CODECS are sent to the multiplexer (mux) on 5 separate buses, as shown in figure 15.

b. The mux simultaneously receives one bit from each of the five even-numbered CODECS every 2.6 microseconds. After one 8-bit PCM word is collected from each even-numbered CODEC, the process is repeated for the odd-numbered CODEC. In this manner, 30 PCM words are received and buffered every 125 microseconds. The words are then routed in serial form at a 2.048-megabits-per-second (Mb/s) rate to the port group highway (PGH) for subsequent transmission to the data conditioner. The multiplexing process occurs continuously. Two separate buffers are used to allow 30 PCM words to be transmitted to the data conditioner while the next 30 words are being input from the CODECS. The buffers output at alternate 125-microsecond intervals to provide a continuous stream of data to the port group highway.

c. The process is reversed for the opposite direction of transmission. The demultiplexer continuously receives and buffers 30 PCM words from the port group highway while transmitting the previous group of 30 words to the CODECS. The demultiplexer outputs to 10 different CODECS on 5 separate buses.

7.02 Digital Transmission Network.

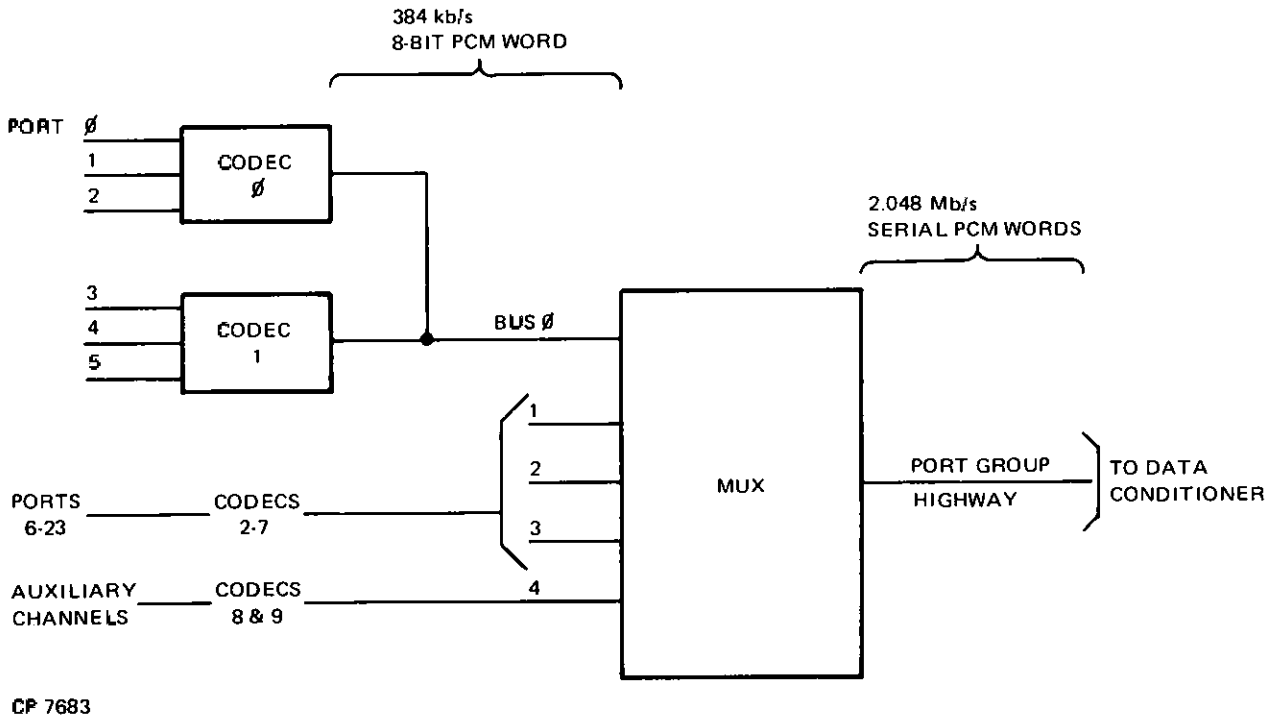
a. The arrangement of the digital transmission network and its relationship to the ports of the DBX is shown in figure 16. One-half of the dual data conditioner and the associated timeslot interchange (TSI) switch can handle up to eight port group highways (PGHs). A PGH carries the duplex information of 24 ports plus 8 auxiliary channels. Each PGH has a total of 32 channels and each half of the dual data conditioner handles a total of 256 channels.

b. The data conditioner (fig. 17) receives incoming differential data signals converting them to TTL signal levels. The serial-in parallel-out (SIPO) function

maintains the 2.048-Mb/s data rate to the TSI switch by combining the serial data from eight PGHs into a single 8-bit parallel data world channel. One 8-bit PCM word from each of the input signal lines is shifted into a set of eight 8-bit shift registers. Upon loading the eighth bit, the contents of the eight registers are broadside-loaded into another set of eight shift registers. The eight words are now bit parallel, word serial. During the next eight clock cycles, the eight words are clocked out to the TSI switch while the next set of eight words is shifted into the input register. The shifting process is synchronized by the DBX master frame pulse. Parallel data coming from the TSI switch gets exactly the reverse treatment. The parallel-in serial-out (PISO) function takes the single 8-bit parallel data channel and separates the data into individual serial data for each of the eight PGHs. The output data is transmitted as differential signals. Data conditioner operation is continuous, whether data is present on the transmission channels or not. The data conditioner is synchronized with the PCM circuits by the master clock.

7.03 Timeslot Interchange.

a. The TSI switch (fig. 18) uses time-division multiplexing (TDM) concepts to switch 256 channels. A set of eight 256 X 1 memories are used on the send side to receive the parallel 8-bit words from the data conditioner, and a set of eight 256 X 1 memories are used on the receive side to store the data for output to the data conditioner. A send and receive memory location is dedicated to each of the 256 channels switched by the TSI. There are 256 timeslots available for timeslot interchange. During the first half of the clock cycle, memory addressing is under control of the timeslot counter. Data from the equivalent port address is clocked into the send memory at this time. During the second half of the clock cycle, memory addressing is under control of the send and receive store memories that transfer data from the send memory to the receive memory in accordance with any programed data in the store memories. The call store memories are programed by the controller, which, in turn, receives its direction from the CPU.



NOTE. 2 PGH Timeslots are not multiplexed.

Figure 15. DBX Data Mux

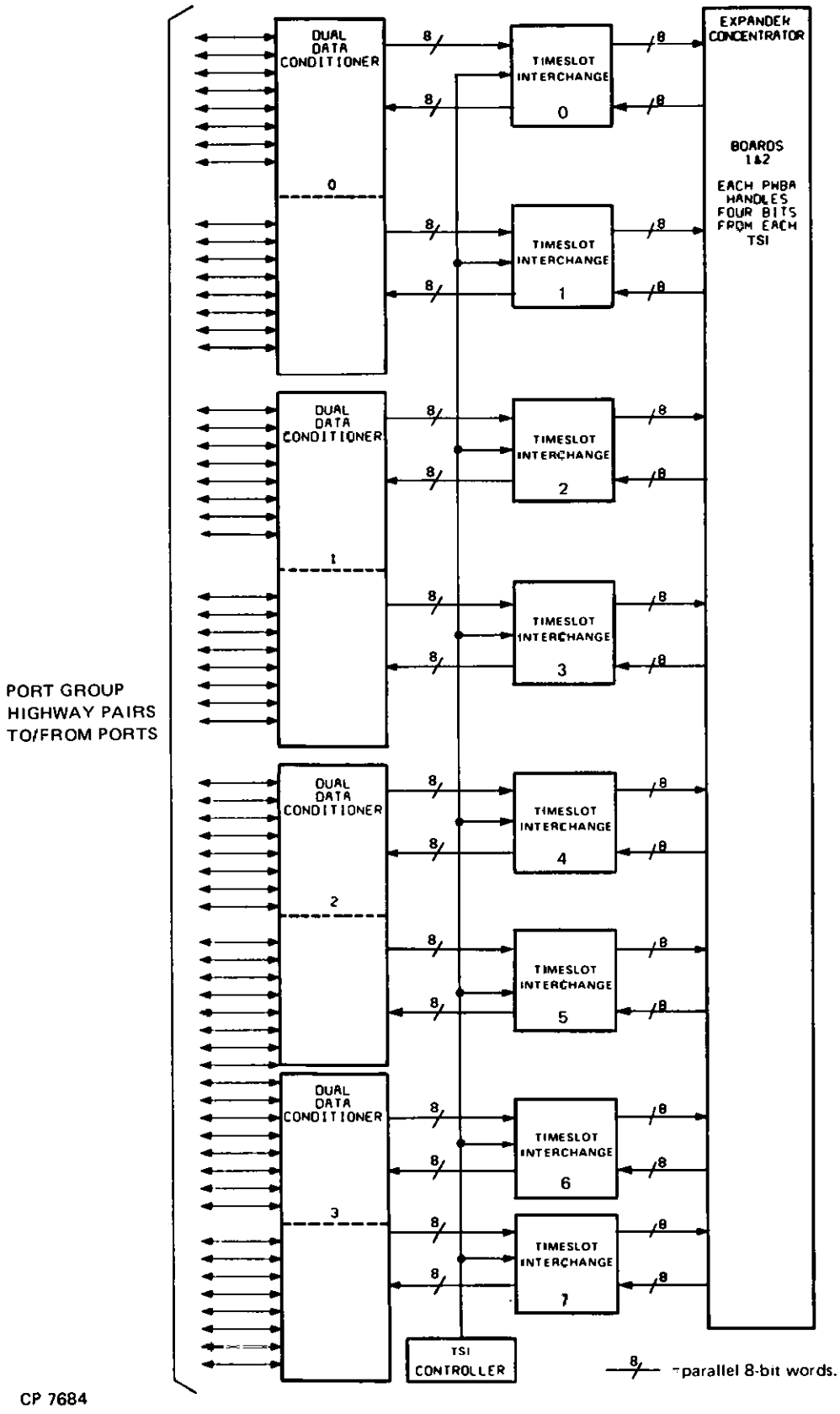


Figure 16. DBX Digital Transmission Network

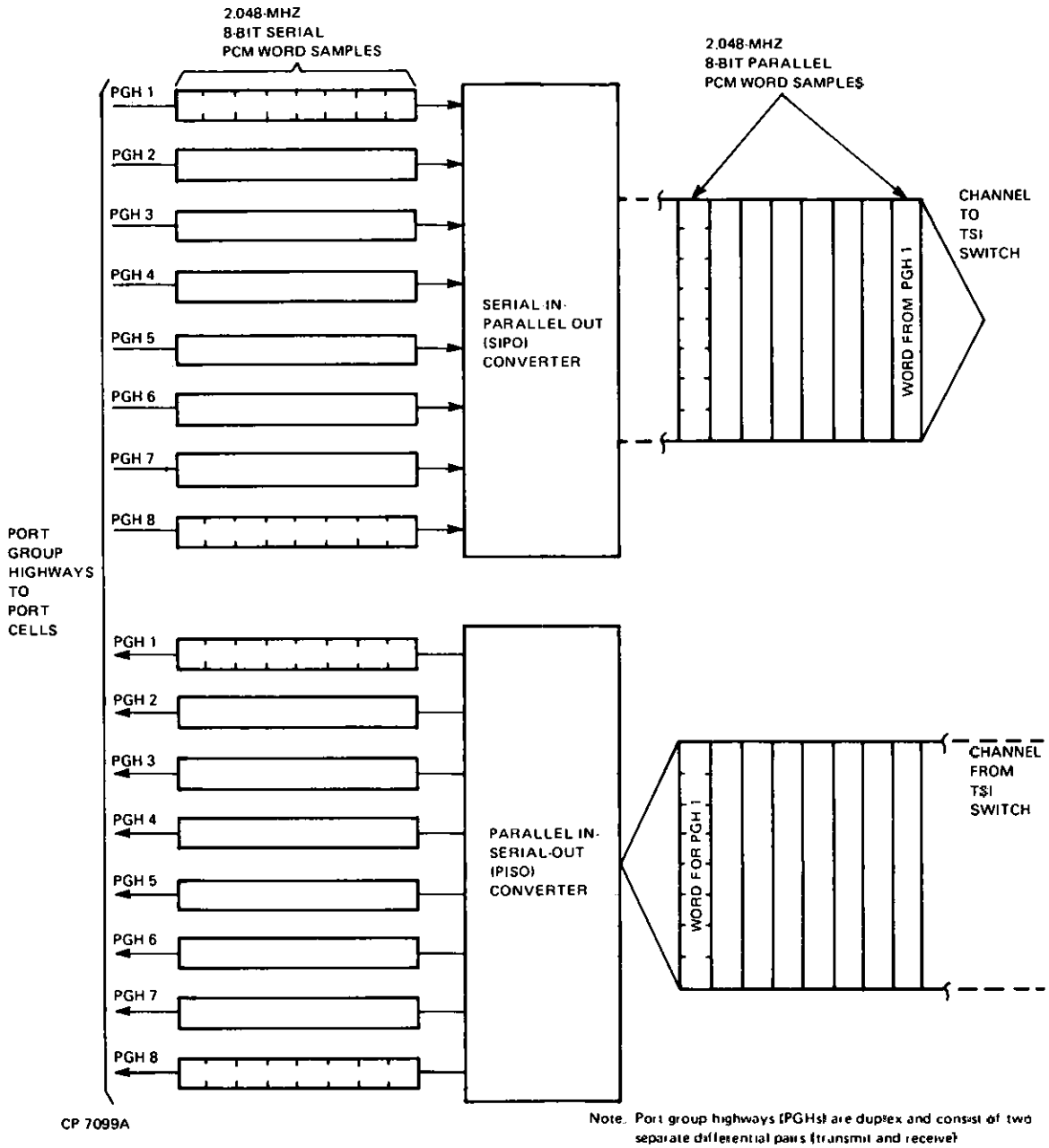


Figure 17. Data Conditioner

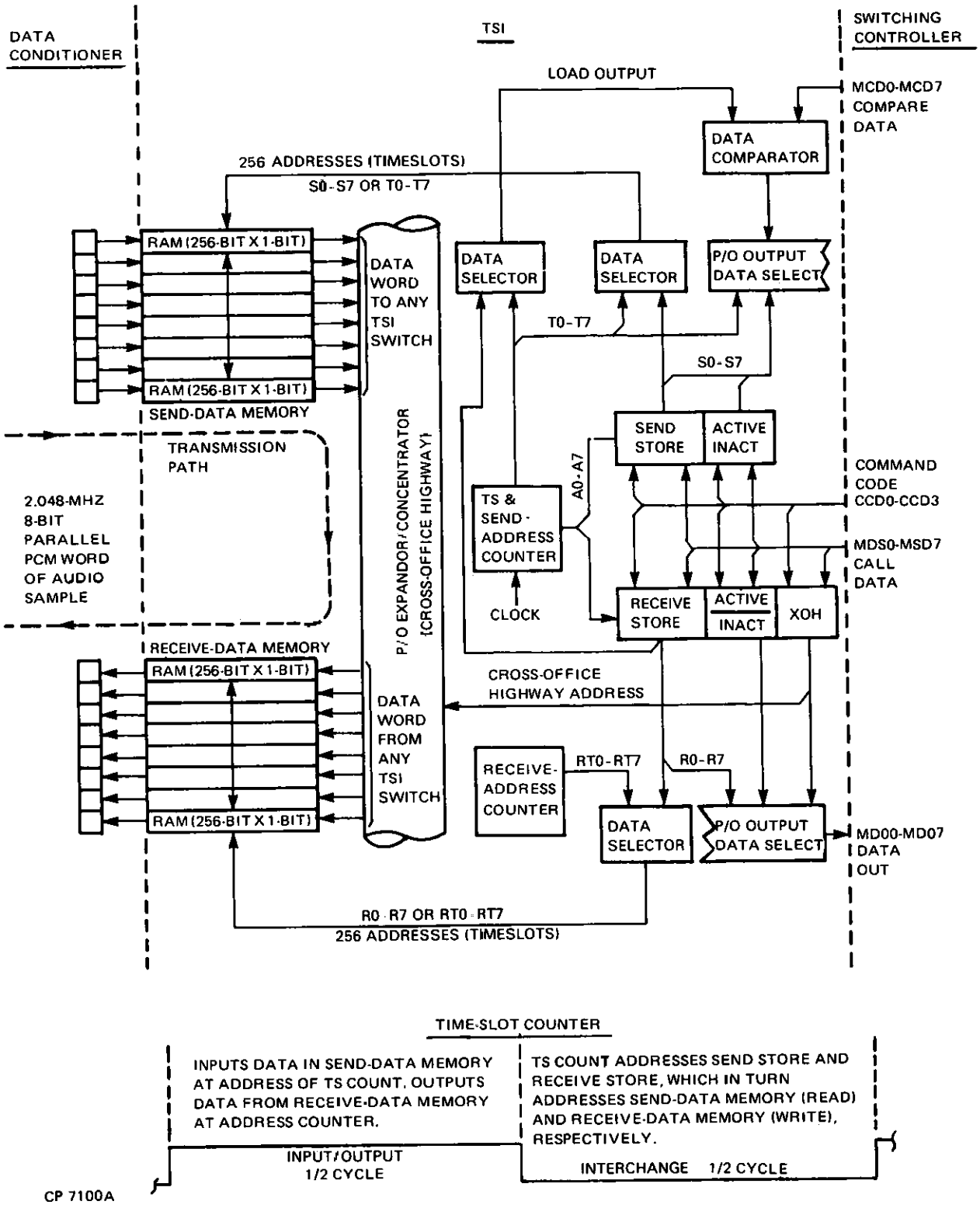


Figure 18. DBX Timeslot Interchange

The call store memory consists of twenty-three 256 X 1 memories. A memory location is used for each timeslot. Eight memories store the send-port numbers, with a ninth memory used to designate whether the timeslots are active or inactive. Eight memories store the receive-port numbers, with a ninth memory used to designate the timeslot's status. Five memories store the cross-office highway numbers that designate transfers. The controller, under control of the CPU, has the capability to write data into any given timeslot in the send or receive store memories, read the data previously stored in any given timeslot, or search the receive-store memory to determine the timeslot a given receive-port number has been stored in. The send active/inactive and receive active/inactive memories provide the ability to store call data in the TSI and reserve timeslots while data transfer is inhibited. Programing inactive call data does not interfere with data transfer on active timeslots. This permits reserving a timeslot for a given port while at the same time actively transferring data with the same port on a different timeslot. To guarantee the integrity of the idle line condition and to prevent transmission of random characters to idle receive ports, a write-after-read memory cycle is incorporated in the receive data memories to restore them to the idle line condition after data is transferred out.

b. The expander-concentrator is a space-divided network that permits up to eight TSI switches to be interconnected. The expander-concentrator consists of combinational logic, and the data is steered by the cross-office highway data supplied from the cross-office store memories.

7.04 TSI Controller.

The TSI controller programs the proper interchange of timeslots on the TDM cross-office highways. The basic traffic decisions affecting timeslot interchange are done by software in the CPU and then relayed to the controller by way of the interrupt encoder. In addition, the CPU can initiate instructions that cause the controller to read selected portions of the TSI call storage memories. The selected data is transferred back to the CPU without affecting timeslot interchange. The controller continuously monitors the control lines from the interrupt encoder to detect initiation of the next command cycle from the CPU. Between command cycles, the interface between the controller and the TSI is static, and no control signals or data are exchanged. The enable signal from the interrupt encoder strobes the call data and command words into the selected TSI registers. The controller then energizes the TSI data buses and enables the selected TSI. After completion of the specified command, the TSI signals the controller it is finished and energizes the output data bus, if required. The controller modifies its status register to indicate the TSI has executed the command and, if required, clocks the data from the TSI into its data-out register. If the TSI does not signal that it has executed the command within a designated time period, the controller modifies its status register to indicate a TSI timeout, or failure to complete. The CPU, by way of the interrupt encoder, periodically polls the controller status register until either the done or error flags are detected. When the CPU detects the done or error flag, the data from the output data register is transferred, when required, and the command cycle is completed.

8. DIGITAL-TO-ANALOG CONVERSION (DECODING)

8.01 Introduction.

On the calling or called party side of a line-to-line call, The D/A process starts with a digital signal from the TSI. This signal is shifted into a 1 x 256-bit RAM in the port group MUX/DEMUX circuit where bit selection and gating logic (demultiplexing process) directs the digital signal to the CODEC in the port group circuit. This signal enters the interface circuit of the CODEC and is converted on an 8-bit parallel form (eight separate lines). The CODEC's digital-to-analog converter (DAC) uses this parallel form and converts to analog form for the CODEC's operational amplifier (OA). The PAM signal from the OA is filtered by a three-stage receive filter and sent to an input/output port circuit. The actual D/A conversion is done at the DAC circuit.

8.02 Shannon-Rack Decoder, Background Information.

C. E. Shannon and A. J. Rack devised a simple decoder to decode a binary number (given in serial form) with the least-significant digit first. The Shannon-Rack decoder recognizes that each pulse of a received code group has a value that is half that of the next pulse. In the decoder scheme, the 2^0-digit pulse of an n-digit number contributes $\frac{K}{2^n}$ volts to the final output of the decoder, the 2^1-digit pulse contributes $\frac{K}{2^{n-1}}$ volts, the

2^2-digit pulse contributes $\frac{K}{2^{n-2}}$ volts, etc. The linear summation of the contribution of all the pulses is a voltage that is directly proportional to the number to be decoded. Figure 19 shows the decoder. A gate circuit is shown as a switch that, in response to the pulses of a code group, is positioned to the left for a predetermined length of time. Whenever the RC network is connected to the constant-current generator, an equal amount of charge is added to the capacitor C, and the potential E changes by a fixed amount. In the interval between pulses, the gate circuit is open (as shown by the switch in figure 19), and the capacitor discharges through the resistor, resulting in an exponential rise in the potential E. The RC time constant is arranged whereby the capacitor charge decreases to one-half of its original value in the time interval between pulses, representing consecutive powers of 2. This interval is a pulse period, P. With the arrival time of the 2^0-digit pulse of an n-digit number corresponding to the start of the time scale, and assuming that the capacitor is initially discharged, a 2^0-digit pulse deposits a charge Q on the capacitor. Because the capacitor discharges to one-half of its original value in each pulse period, the original charge decreases to $\frac{Q}{2^n}$ at time T, where T = nP. Successive pulses contribute $\frac{Q}{2^{n-1}}$, $\frac{Q}{2^{n-2}}$, etc. to the capacitor charge at time T. The potential E continues to rise exponentially after the nth pulse, and, hence, there is only one instant at which E is correctly related to the binary number to be decoded. For this reason, the sampling circuit must be pulsed at exactly T = nP. To overcome the need for precise timing of the pulse applied to the sampling circuit, A. J. Rack added an underdamped RLC circuit in series with the RC network. With this network, the waveform of E is made flat at time T, and there is considerable time interval during which the proper potential is applied to the holding capacitor by pulsing the sampling circuit.

8.03 PCM Code Selection Criteria.

There are three fundamental operations in a PCM system that can influence the choice of code: encoding, digital transmission, and decoding. Any given principle or mechanism of encoding, such as counting, weighing, matrix translation, etc., imposes restrictions on the type of code generated. The encoding and decoding mechanisms impose restrictions on the order in which characters are allotted to levels, rather than on the set of characters used. The transmission path frequently limits the sequence of digits that can be transmitted, and one effective way of restricting the sequence is to restrict the set of characters. Since the restrictions imposed by encoding, transmission, and decoding are independent, they can easily be inconsistent. Thus, the need can arise to generate a certain code – called code A – simply because it is convenient for the encoder. It may be, however, unsuitable for line transmission, so it is translated by digital processes into code B, which is transmitted easier. Additionally, if neither A nor B has a simple decoding mechanism, then a further digital translation to code C is required.

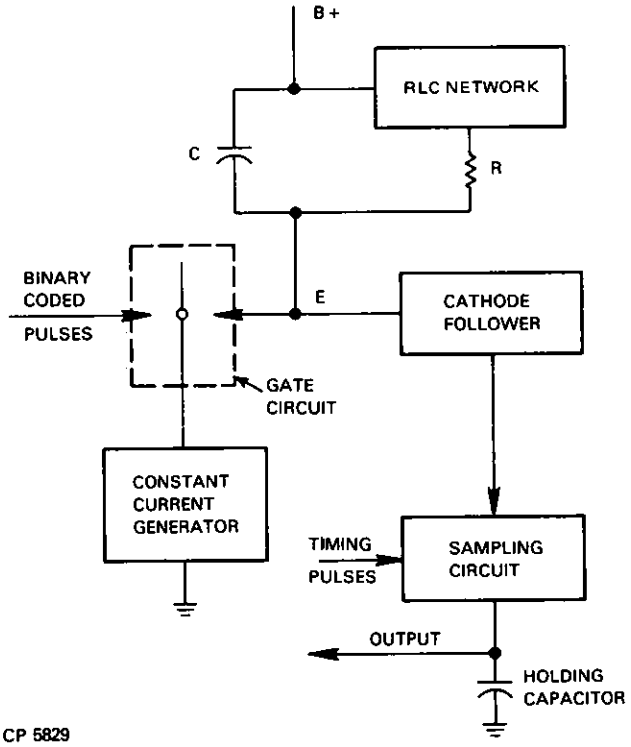


Figure 19. Shannon-Rack Decoder