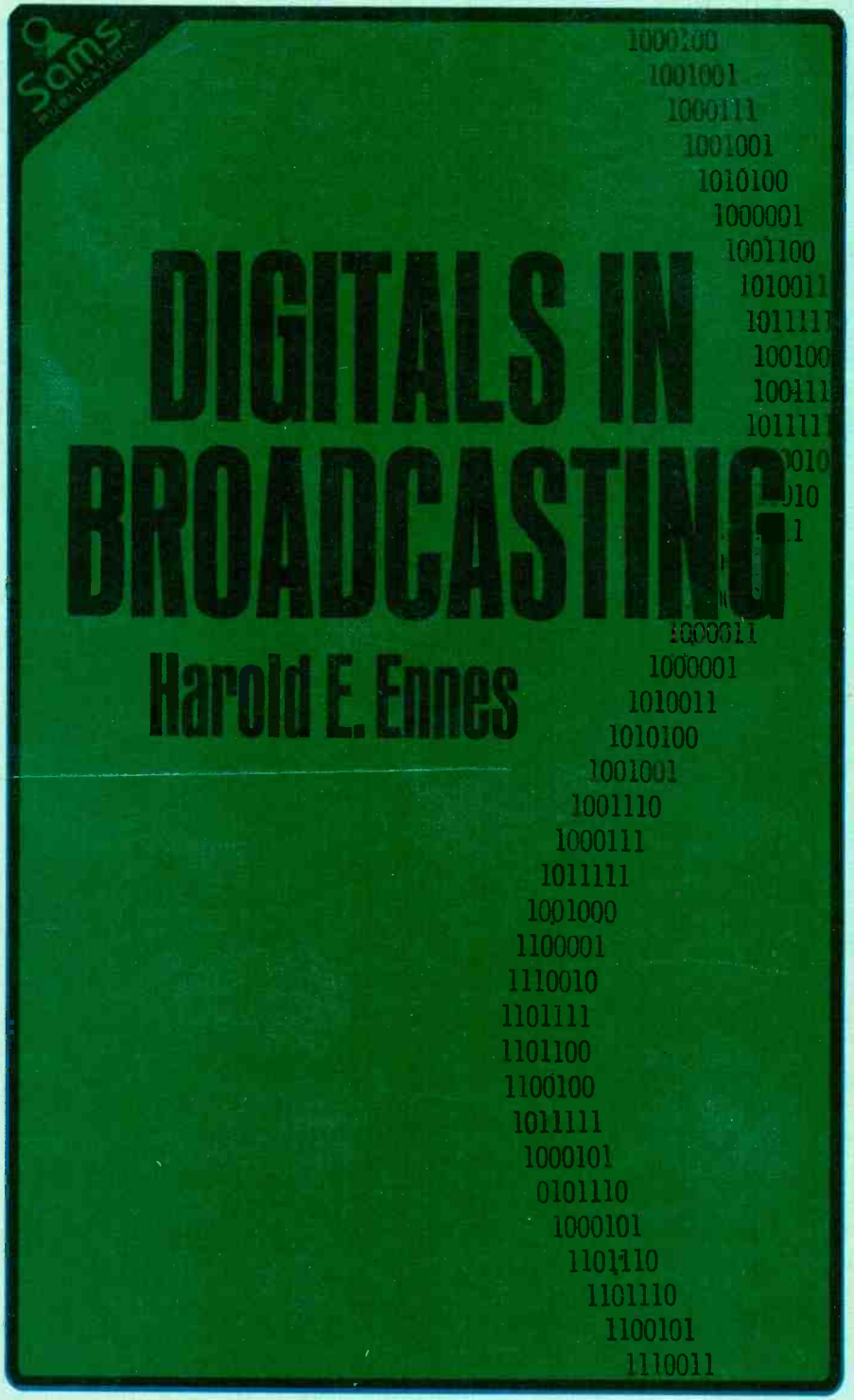


DIGITALS IN BROADCASTING

Harold E. Ennes

Digitals in Broadcasting

by

Harold E. Ennes

Howard W. Sams & Co., Inc.
4300 WEST 62ND ST. INDIANAPOLIS, INDIANA 46268 USA

FIRST EDITION
FIRST PRINTING—1977

International Standard Book Number: 0-672-21414-8
Library of Congress Catalog Card Number: 77-73922

Printed in the United States of America.

Preface

Digital systems as considered in this book are currently being used in two broad areas of application: (1) digital control of analog systems and (2) digitized audio and video equipment for specific purposes. Digital control has been with us for some time, but it is still not thoroughly understood by many in the communications field. Digitized audio and video is an entirely new field at the time of this writing, and it promises to revolutionize system concepts. Digital processing of analog signals can perform many tasks that are difficult or impossible to achieve by strictly analog techniques.

Like stereo in fm and color in tv, digitals are emphasizing the continual need for technicians to update their knowledge. Unfortunately, because of the rapid influx of digital devices into the broadcasting, recording, and general communications fields, students and practicing technicians have been faced with a scarcity of necessary background information relative to digital technology. *Digitals in Broadcasting* has been written to meet the need for a practical source of such information.

This text is *not* a rigorous treatment of the field of digitals. Rather, it is slanted to the practicing technician (or student) who has not had formal computer training. A simple building-block approach is used to provide a practical and effective marriage of analog and digital worlds in the mind of the reader who has had little or no exposure to digital technology. As such, the treatment is as basic as possible, while providing the necessary engineering background to enable a technician to feel more at home with the instruction books that accompany modern digital equipment. Computer training, in itself, is not geared to the specific needs of the broadcast and general communications aspects of digital technology. While computer-type equipment is involved, the practical applications are different. I have, therefore, deliberately bypassed many details relative to spe-

cific computer applications and have emphasized only those characteristics of analog-digital functions most important to the intended field.

Besides serving as a basic reference, this book will provide a quick refresher course for the more advanced digital technician. A basic electronics background is assumed.

It is my sincere wish that this text not only will serve as an effective introduction to digital techniques, but also will inspire the reader to more advanced studies in the field. Equipment will change with new discoveries and improved techniques; however, the fundamental steps taken in this book will serve as a solid foundation for a continued training program. Comments and constructive criticisms by readers are welcomed.

Acknowledgments: A technical book carrying a single author's name is almost always the result of research and development from many different sources: individuals, companies, and technical organizations. This is certainly true of this book. My purpose has been to coordinate basic theory with the original research and development accomplished by others, and to organize this material into an effective and practical training program for the intended reader.

My special thanks to Frank Thompson, J. Diermann, M. O. Felix, Mark Sanders, and Charles Anderson, all of the Ampex Corporation, for their invaluable assistance.

My sincere appreciation is extended to the following companies, corporations, and organizations for their contributions that have made this book possible: Ampex Corp.; Consolidated Video Systems, Inc.; Datavision Video Products Div., 3M Co.; Electronic Engineering Co. of California; Hewlett-Packard; Ikegami Electronics (USA), Inc.; Lexicon, Inc.; Moseley Associates, Inc.; Oliver Audio Engineering; Philips Broadcast Equipment Corp.; SMPTE Journal; TeleMation, Inc.; and Visual Electronics Corp.

HAROLD E. ENNES

Contents

CHAPTER 1

CHAPTER 2

CHAPTER 3

CHAPTER 4

CHAPTER 5

CHAPTER 6

CHAPTER 7

CHAPTER 8

CHAPTER 9

CHAPTER 10

CHAPTER 11

APPENDIX A

APPENDIX B

APPENDIX C

CHAPTER 1

Orientation in Bits, Bytes, Bauds, and Bandwidth

As the term *digital* implies, digital technology involves "doing it by the numbers." In digital technology, the accuracy is determined in the initial system concept; the "readout" is in no way subject to an operator's estimate.

There are two basic applications for digital systems:

1. *Digital control circuitry.* This is already well established in the field, but the penetration into broadcasting has been so broad and so rapid that the subject is not thoroughly understood by many.
2. *Digital data (information) transmission,* such as the conversion of analog (continuously varying) audio, video, or other information to digital (number) form. This is a new and rapidly developing field.

The purpose of Chapter 1 is to give an overview of these digital applications in broadcasting, with implications in the sound-recording and general communications fields. It is like stepping back from a painting in order to appreciate the beauty: Rather than going directly into a complex system analysis, we will first step back and take a look at the entire philosophy of digital and digital-analog technology.

One of the first hurdles to clear is the mystery of terminology. There is nothing so frustrating as reading a text full of terms that do not appear at all in a regular dictionary but require a special glossary or dictionary of scientific terms. After learning the terminology, you are ready to view the overall picture more realistically.

Table 1-1 lists standards prefixes as used in this text, along with the most common applications.

Table 1-1. Prefixes

Prefix	Definition	Power of Ten Multiplier
atto	millionth of millionth of millionth part	10^{-18}
pico (preferred)	millionth of one millionth part	10^{-12}
micromicro (obsolete)	millionth of one millionth part	10^{-12}
nano	thousandth of a millionth	10^{-9}
micro	millionth of one part	10^{-6}
milli	thousandth of one part	10^{-3}
centi	hundredth of one part	10^{-2}
deci	tenth of one part	10^{-1}
deka	ten	10^{1}
hecto	one hundred	10^{2}
kilo	one thousand	10^{3}
mega	one million	10^{6}
giga	one billion	10^{9}
tera	one trillion	10^{12}

Most commonly used letter symbols:

ns = nanosecond
μs = microsecond
ms = millisecond
μF = microfarad
pF = picofarad
kHz = kilohertz*
MHz = megahertz*
GHz = gigahertz*

* 1 Hz = 1 cycle per second

1-1. INTRODUCTION TO THE BINARY SYSTEM

You have been working with digitals all your life. Your usual method has been "base 10" arithmetic (also termed *decimal* arithmetic). This means that the columns under which you start your new 1s are headed 1, 10, 100, 1000, etc. The decimal system has ten digits, 0 through 9. This system is handy, because you first learned to count on your digital appendages, 10 fingers and 10 toes. But you will learn to interface this system with a system that recognizes only two numbers, 0 and 1. This is *binary*, or base-2, arithmetic.

You will be surprised how easily the binary system can be learned and put to use. When you become familiar with it, the math is not as involved as that required for basic electronics. For example, binary logic can say that 1 = true and 0 = false. Or 1 is "high" and 0 is "low." Or 1 = + and 0 = 0 (ground). Or 1 = 0 and 0 = minus. Or 1 = Jill and 0 = Jack.

The binary system is the simplest possible number system. A single digit in the decimal system can be any digit from 0 to 9. A single binary digit (called a *bit*, as a contraction of *bi*nary digi*t*) can be only a 0 or a 1 (in electronic parlance, a "pulse"). This is extremely

valuable in electronic circuitry because a 0 can be an open switch (no current) and a 1 can be a closed switch (maximum current). Thus, a single bit has just two possible *levels*, 0 or 1.

Just as the base 10 has ascending powers of 10, the binary system has ascending powers of 2: 2, 4, 8, 16, 32, etc. Note an important point: misplacing the decimal point by one place results in an error of a power of ten; misplacing the binary point by one place results in an error of only a power of two. In addition to this, since the binary system employs only zeros and ones, a technique known as a *parity check* can be used to detect an error. Finally, there are *error correction codes* (Chapter 5) that make the binary system virtually foolproof, except from human errors in initial programming.

Since a single bit is only a 0 or a 1, *groups* of bits are required to define any *number* greater than 1, or any *character*, such as letters of the alphabet, teletypewriter signals, signs, symbols, etc. This group of bits is termed a *byte*. The number of bits in a byte depends on the total number of characters required for the entire code being used. For example, if you have a total of 128 numbers and characters to be defined, you need a 7-bit byte ($2^7 = 128$).

A *word* in computer parlance is an ordered set of characters which depends on the specific system design. For example, a digitally controlled color camera (Chapter 9) employs a 24-bit command word. Since up to six cameras can be controlled, the first 3-bit byte addresses the particular camera to be commanded in the immediately following part of the word. This is followed by a 15-bit byte of simultaneous commands, and then by six bits of one-at-a-time commands.

In the above example, transmission bit-by-bit is done on one wire (serially). In some systems, you can also have parallel feed, one wire for each bit. Thus, eight wires would be required for eight bits. Or, you might have 24 wires with three groups of eight bits on each wire. This 24-bit group may be called a word. Each 8-bit subgroup is called a byte. In general, the terminology is spelled out for each specific system. In some systems, the group of bits forming a byte is termed a *binary word.*

1-2. SIGNALING SPEED VERSUS BITS PER SECOND

Digital transmission rates may be expressed in terms of *bits per second* (sometimes abbreviated bps) or *bauds*. These terms do not mean the same thing. The term *baud* is used as a measure of *signaling speed.* The term *bits/second* is used as a measure of the number of *information* bits per second. Let us clarify this now.

Assume you have the pulse train illustrated in Fig. 1-1. Ten pulses occur each second, so the frequency for the indicated train of pulses

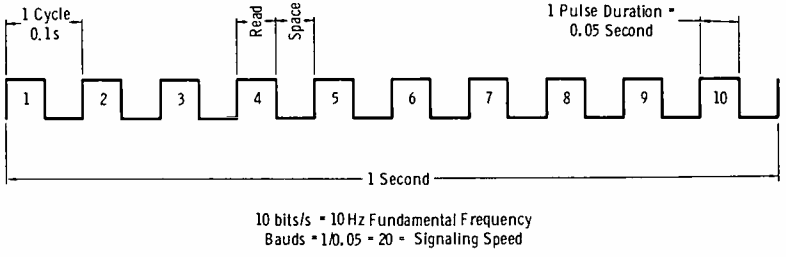


Fig. 1-1. Bits and bauds in a pulse train.

is 10 Hz. The pulse repetition interval is $1/10 = 0.1$ s. Thus, you have $1/0.1 = 10$ bits/s. Assuming spaces equal to the pulse duration, the width of each pulse is $\frac{1}{2}(0.1) = 0.05$ s, or 50 ms.

A baud is the *reciprocal of the length of the shortest signal element*. So:

$$1/0.05 = 20 \text{ bauds} = \text{signaling speed}$$

The fundamental frequency is the reciprocal of the pulse repetition interval, or:

$$1/0.1 = 10 \text{ Hz}$$

Note that when the spaces are equal to the pulse width (50% duty cycle, or square wave), a simple 2/1 relationship exists between *bauds* and *fundamental frequency*. Also, if each pulse (read period) carries one information bit (no synchronizing pulses such as start-stop), a simple 2/1 relationship holds between *baud rate* and *bit rate*. It is assumed here that the "spaces" do not carry information. An example is the electronic pulse counter.

You can see that the term *baud*, since it is defined as the reciprocal of the shortest signal element, is somehow related to bandwidth. In practice, the required bandwidth is related to the desired bits/second transmission speed (channel capacity), signal power, and noise. Thus, the baud specification is not particularly useful to the practicing engineer or technician. It is explained here only because a specification for a given system may be given in terms of bauds, and you should know what the terminology means.

NOTE: Bits/second *vs* bandwidth is covered in Section 1-4.

Please note that these examples assume a return-to-zero pulse format. This and other formats are covered in Section 1-3.

Some systems, such as the Teletype code, employ start and stop pulses in addition to character *information* pulses. Assume as an example that the start pulse is 25 ms wide, the five information bits are each 20 ms, and the stop pulse is 30 ms. The five information bits

are required to make up a single *character*. The total of 7 bits per character is termed the *unit code*.

The baud rate is:

$$1/0.02 = 50 = \text{signaling speed in bauds}$$

The total time for a character transmission must include the information bits plus the start and stop pulses. So:

$$\text{Start pulse} = 25 \text{ ms}$$
$$5 \text{ information pulses} = 5(20) = 100 \text{ ms}$$
$$\text{Stop pulse} = 30 \text{ ms}$$
$$\text{Total} = 25 + 100 + 30 = 155 \text{ ms for one character}$$

Then the bit rate is:

$$(1/0.155) \times 5 = 6.45 \times 5 = 32.3 \text{ information bits/second}$$

Now assume that a code format requires the insertion of a 40-ms *space* period between the stop pulse for one character and the start pulse for the next. The baud rate would remain the same, since the duration of the shortest bit is still 20 ms. However, the *rate of information flow* would be *reduced*, since the total time for one character is now 155 ms + 40 ms = 195 ms. Then:

$$(1/0.195) \times 5 = 5.13 \times 5 = 25.6 \text{ information bits/second}$$

SUMMARY: The baud rate determines the type of channel required for transmission. Bits/second describes the rate of information flow. When parity bits are used, they are counted the same as information bits.

It is important to note that the rate of information flow is not necessarily the same as the total bits/second used in a given channel. The *channel capacity* (Section 1-4) must consider the total number of bits, or unit code, transmitted per second. In the last example above, there were a start pulse, five information pulses, a stop pulse, and a space pulse. This was a total of 8 pulses which required a time interval of 195 ms. Therefore, the *total* number of pulses sent was:

$$(1/0.195)\ (8) = 5.13\ (8) = 41.0 \text{ pulses/second}$$

On the other hand, if all eight pulses carried message information (including any parity bits), the rate of information flow and the total bits/second would be identical.

1-3. PULSE FORMATS

A summary of the most common pulse formats is shown in Fig. 1-2. A binary value of 101100 (read "one-zero-one-one-zero-zero")

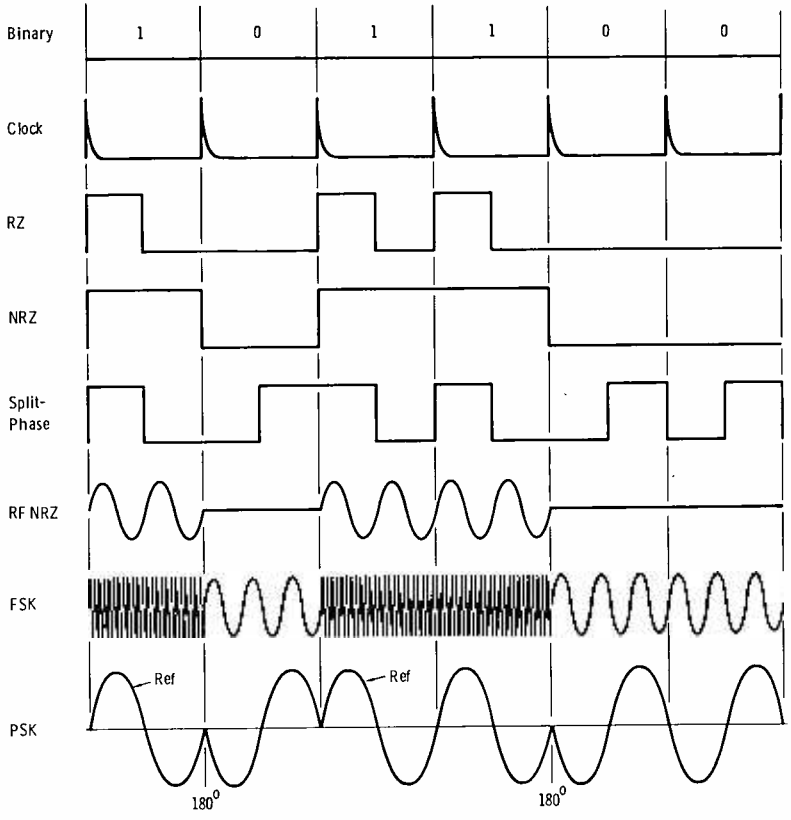


Fig. 1-2. Summary of most common pulse formats.

is listed on the top line as an example. The second line shows the sampling time, which may be either the leading or the trailing edge of the clock pulse. The clock pulse is the main timing pulse for a synchronous system.

RZ. This is the return-to-zero bit representation for binary 101100. This is the format discussed in Section 1-2.

NRZ. This is the nonreturn-to-zero format for the data indicated. Note that a 1 pulse is wider in this code than in the corresponding RZ format, and that two consecutive 1's employ just one (wider) pulse (does not return to zero). The baud rate is about one-half that of the RZ code, requiring about one-half the bandwidth for a given message. Another way of saying this is that for a given bandwidth, the NRZ format can handle a higher data rate than the RZ code.

Split-Phase. This format has a low-to-high transition for each bit, whether 0 or 1. A 1 has a transition from high to low in the middle

of a bit (½-bit duration). A 0 is signified by a low-to-high transition in the middle of a "mark" (sampling time). This is an excellent format for synchronization and accuracy, and it can be self-clocking. The special SMPTE time edit pulse (Chapter 5) is a biphase adaptation of this format.

RF NRZ. This is a straight pulsed carrier wave. Carrier represents 1, and lack of carrier represents 0. If RZ is used, the carrier is pulsed on for each bit with appropriate spaces.

FSK. This is *f*requency-*s*hift-*k*eying in which 1 is represented by a higher deviation frequency and 0 by a lower deviation frequency.

PSK. This is *p*hase-*s*hift *k*eying in which 1 is a leading phase deviation limit and 0 is a lagging phase deviation limit. In the example shown, a complete 180° phase inversion exists between 1 and 0.

NOTE: Fsk or psk may involve either modulated tones or modulated carriers.

In one example of psk, a 950-MHz carrier is used. An audio tone at 5.4 kHz is phase-shifted 180° to identify a single bit of information. This tone is used to modulate the carrier.

Polar and bipolar pulses are discussed in Chapter 8.

1-4. BASIC DIGITAL SYSTEM THEORY

A digital data signal changes only in discrete levels, rather than in a continuously varying (analog) manner. Furthermore, the data are usually binary so that the discrete steps are limited to two, 0 and 1.

There are two basic types of digital systems, *baseband* and *modulated.* A baseband signal is the direct-coded data signal sent over a transmission path without modulation. A pulse-modulated signal is a carrier modulated in one or more modes (Chapter 8).

A binary signal is seldom a pure rectangular waveform, particularly in transmission systems. Fig. 1-3B illustrates that a bandwidth-limited rectangular wave has the same information (0 or 1) as the original signal (Fig. 3-1A). An extremely wide bandwidth is necessary to pass a fast–rise-time pulse without distortion. Bandwidth limiting (such as by a linear-phase filter) eliminates the need for extreme bandwidths. You should understand that the digital pulses may be limited in bandwidth by transmission facilities, but the gating waveform, regenerated at the receiving terminal from transmitter information, is as nearly ideal in waveshape as possible.

The binary system is normally gated (Fig. 1-3C) only at the time necessary to determine the 0 or 1 condition. This eliminates all noise except that which occurs simultaneously with the pulse

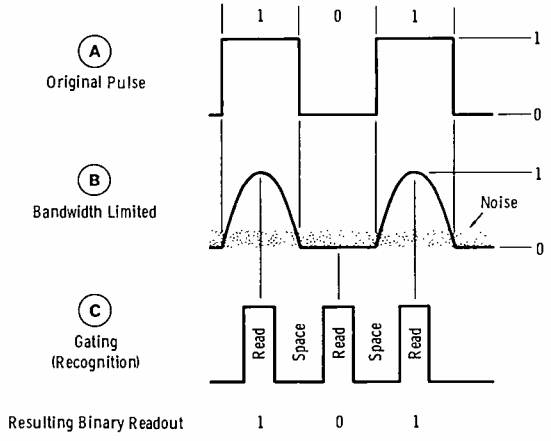


Fig. 1-3. Bandwidth limiting and gating in binary system.

level. Note that noise above a given threshold in the binary 0 position could trigger a false 1. This is treated later in this section.

A binary channel carries a stream of pulses that simply represent zeros or ones. The *capacity* of any binary channel (usually measured in bits/second) depends on the channel bandwidth, B (in hertz); the *average* signal power, s; and the *average* noise power, n:

$$C = B \log_2 (1 + s/n)$$

where,

C is the channel capacity in bits/second,

B is the bandwidth in hertz,

s/n is the signal-to-noise ratio (power basis).

Note that for binary digits (bits), the logarithm to the base 2 is used rather than the more conventional logarithm to the base 10. Some calculators have a means of finding the logarithm to *any* base. Log_2 tables are not readily available. However, base 10 (common) logarithms can be used, in which case the answer is in *hartleys*. This answer multiplied by 3.3 gives the answer in bits/second. We will solve an example in both ways so that you can use this relationship.

NOTE: For a close estimate when extreme accuracy is not required (usually the case), you can use the graph given in Appendix C to find $\log_2$ of any number between 2 and 10,000. Or, simply find $\log_{10}$ of the number, and multiply by 3.3.

Assume a channel with a bandwidth of 2500 Hz (near the upper limit of ordinary unequalized telephone lines). This channel has an s/n ratio of 30 dB. What is the maximum rate at which data can be transmitted on this channel?

To get the answer directly in bits/s, $\log_2$ is used:

$$C = 2500 \log_2 (1 + 1000)$$

NOTE: 30 dB = 1000/1 power ratio.

$$C = 2500 \log_2 (1000)$$

(The 1 can be dropped.)

$$C = 2500 (9.9)$$
$$= 24{,}750 \text{ bits/s}$$

Now solve by the $\log_{10}$ method:

$$C = 2500 \log_{10} (1000)$$
$$= 2500(3)$$
$$= 7500 \text{ hartleys}$$

To convert from hartleys to bits/second:

$$7500(3.3) = 24{,}750 \text{ bits/s}$$

Remember that this formula gives the *absolute maximum* (theoretical) information rate for a given channel with uniform random noise. The relationship of channel capacity, bandwidth, and s/n ratio is derived from *Shannon's theorem,* and it is used by engineers to determine the theoretical limits of one unknown value from the two known variables.

IMPORTANT: To comprehend channel capacity correctly, remember that this includes all 1's and all 0's (yes-no decisions). It *does not* mean just the 1 bits.

Before leaving the subject of logarithms to the base 2, remember that a logarithm is an exponent. When you are working with direct powers of 2 (Table 2-3, Chapter 2), $\log_2$ is evident. The number of information bits (I) required to identify a particular selection from a group of n possible selections is $\log_2 n$:

$$I = \log_2 n$$

For example, if you have eight characters to be recognized, $\log_2 8 = 3$. Note from Table 2-3 that $2^3 = 8$. This simply means that the power of 2 is the $\log_2$ of 8, and three bits are required to define eight characters.

A common communication code contains 128 different characters. How many bits must it have per character? Answer: $\log_2 128 = 7$ bits ($2^7 = 128$). This is evident from Table 2-3.

Also note that $2^9 = 512$ and $2^{10} = 1024$. This means that $\log_2 512 = 9$ and $\log_2 1024 = 10$. In the previous example, you could have closely estimated the value of $\log_2 1000 = 9.9$ from this relationship.

Note two important points that can be derived from Shannon's relationship:

1. A message occupying bandwidth B1 can be sent over a channel of lesser bandwidth B2 if the signal power is increased (greater s/n ratio).
2. A given channel capacity (C) can be maintained with a lesser s/n ratio if the bandwidth is increased.

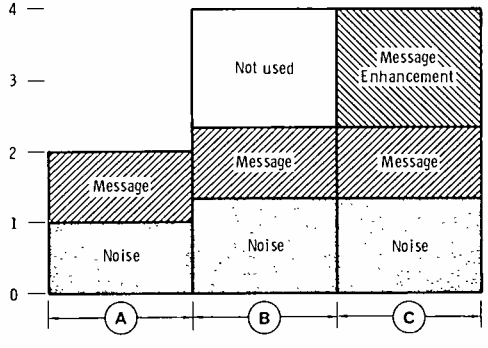
You may wish to disagree with the latter statement, since noise power increases with bandwidth. However, the noise increases only as the *square root* of the increase in bandwidth. Thus, if the bandwidth is doubled, the noise increases as the square root of 2, or 1.4 times. A bandwidth increase of 9 times increases the noise 3 times.

Now let us see what increased bandwidth really means in a digital system. Observe Fig. 1-4. In A, the system has an arbitrary bandwidth of 2 units. Since the noise is 1 unit and the signal (message) is 1 unit, s/n = 1.

In B and C, the bandwidth is doubled. The noise increases to 1.4 times its value in A. Since the signal has the same value in B as in A (1 unit), the s/n ratio deteriorates to 0.7 times that in A. The remaining capacity of the increased bandwidth is not used in B.

In C, the extra bandwidth is used for *message enhancement*. This means that the increased capacity is used to enhance the digital information by some form of *redundant* transmission. You can say that each character is "explained" much as you do on a noisy telephone or communication channel. Here you can see that the s/n ratio is increased to 1.8 times that in A.

Message enhancement takes many forms, such as multiple parity bits, special codes (Chapter 5), actual redundant transmission (repetition), or complement parity.



	B	n	s	s/n
A	2	1.0	1.0	1.0
B	4	1.4	1.0	0.7
C	4	1.4	2.6	1.8

where, B = bandwidth
n = noise
s = signal (message)
s/n = signal-to-noise ratio

Fig. 1-4. Improvement of digital system with increased bandwidth.

The *complement* of a binary word is simply its inverse value; all zeros are changed to ones, and ones are changed to zeros (Chapter 2). For example, a 16-bit command word may be followed by a 16-bit parity that is the complement (interchanged 1's and 0's) of the preceding command word. Unless the command word is followed by its complement, nothing happens, and the code is repeated. This provides reliable random or impulse noise immunity, and it virtually eliminates false commands and erroneous functions in the presence of noise. Review Fig. 1-3 and note how effective this could be when noise occupies a considerable portion of the 0 level, triggering a false 1.

Note also from Shannon's relationship that the minimum s/n power ratio necessary for a binary channel of known bit rate and bandwidth can be found. The simplified relationship in terms of s/n ratio is:

$$s/n = 2^{C/B} - 1$$

For example, assume you have a 2500-Hz channel with a bit rate of 10,000 bits/second.

$$\begin{aligned} s/n &= 2^{10,000/2500} - 1 \\ &= 2^4 - 1 \\ &= 16 - 1 = 15 \end{aligned}$$

Thus, the minimum power ratio must be 15/1. To convert a power ratio of 15 to decibels:

$$\begin{aligned} dB &= 10 \log 15 \\ &= 10\ (1.176) \\ &= 11.76, \text{ or } 12 \text{ dB minimum} \end{aligned}$$

The problems concerning random (white) noise and impulse noise are explored in Chapter 8.

1-5. AUTOMATION

Fig. 1-5 is a block diagram of a typical automation system for a television studio. The heart of such a system is the *digital control unit* (*dcu*). This unit includes the *clock*, a master timing device used to provide the basic sequencing pulses for operation. It also incorporates a high-capacity *memory* (storage) *bank*.

The clocked storage display unit is a memory register (Chapter 6) that automatically records the progress of real time (or some approximation to it), records the number of operations performed, and provides a visual display of contents yet available to the auto-

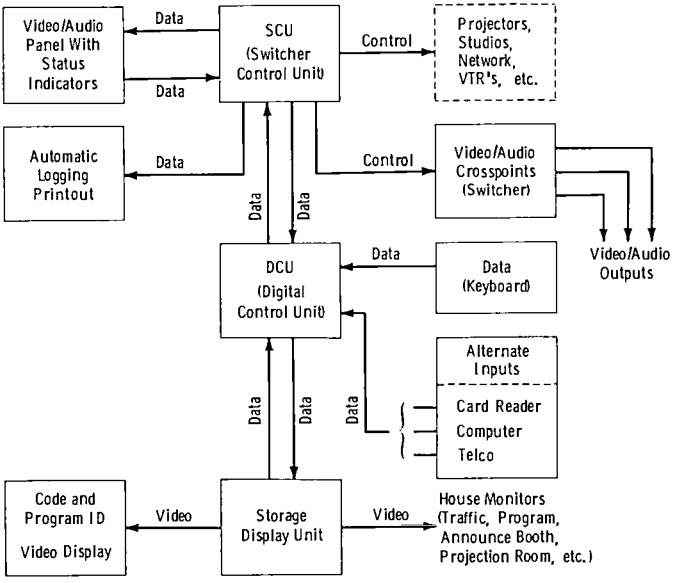
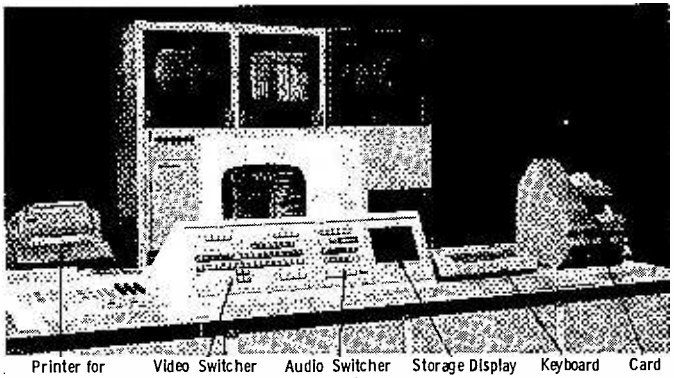


Fig. 1-5. Typical automated control of tv studio.

mated segment of programming. Note that this involves conversion of digital data to a video signal. Fig. 1-6 illustrates an automated switcher system, showing the display on a video monitor.

The switcher control unit (scu) prerolls and "takes" film projectors and video tape recorders (vtr's), and it "takes" selected studios, remote sources, or network sources. It also interrogates the dcu memory and receives data concerning characters and their loca-



Courtesy Visual Electronics Corp.

Fig. 1-6. An automated tv switcher.

tions, compares the data with information in the scu memory, and then operates the appropriate relays or electronic crosspoints in the video/audio switcher. The scu also inserts data into the dcu memory to perform the countdown function, the upshift function as events go on the air, and the forward count of the true-time indication. Bypass of automation to allow manual operation from the video/audio panels is always provided in such systems. Automated controls are covered in Chapter 9.

1-6. REMOTE CONTROL OF TRANSMITTERS

Remotely controlled transmitters are commonplace in the broadcasting and general communications fields. The digital system, due to its extreme reliability and immunity from the effects of noise, has become the most prevalently used system.

Fig. 1-7 represents a typical remote-transmitter control system employing digital techniques. *Telemetry* is a process by which measurement of a quantity is transferred to a remote location to be displayed or recorded, or to actuate a process. The type of telemetry depends on whether the interconnection link is a land line (as through Telco facilities) or an rf carrier or subcarrier between sending and receiving points. In the latter case, fsk or psk is often used; tones (or modulated tones) are used with land lines.

The *multiplexer* allows multiple channels of information to be transmitted and received on a single wire pair or rf channel. It may work on the frequency-division or time-division principle.

A typical basic system may provide 32 channels. This allows 32 parameters to be displayed. With any such system, the number of command functions is twice the number of display functions, since a "raise" and "lower" command are required for each parameter

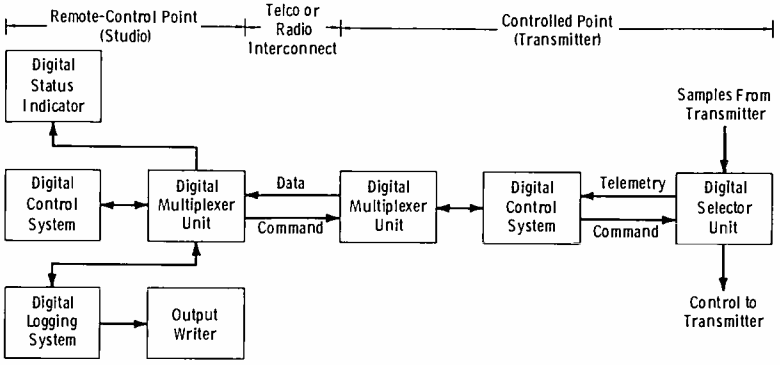


Fig. 1-7. Typical transmitter remote-control system.

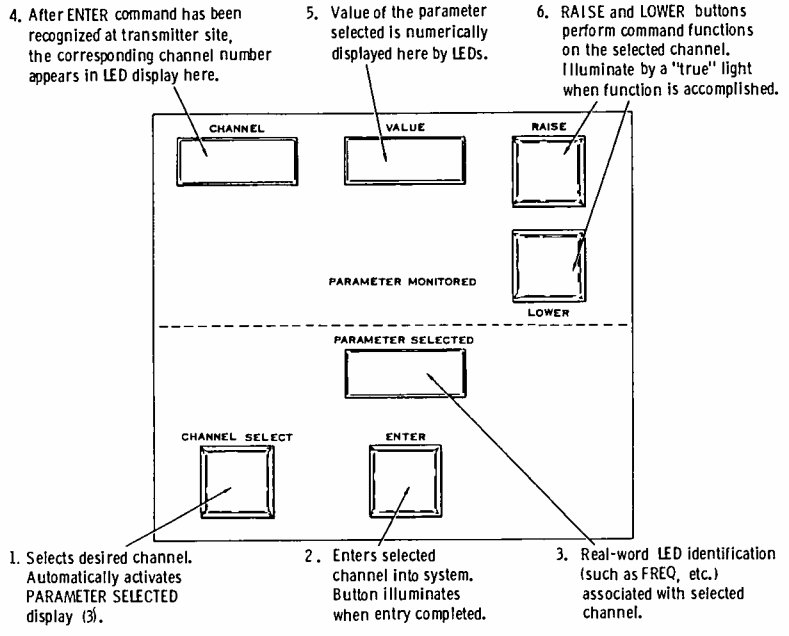


Fig. 1-8. Typical indicating control panel at studio for remote control of transmitter.

monitored. Many systems can be expanded to as many as 96 channels, with 192 individual command functions.

Fig. 1-8 represents a typical indicator and control panel installed at the studio for remote control and monitoring of the transmitter. Follow the numbers from 1 to 6 for a description of the way the system works.

Bipolar pulses (Chapter 8) are used so that the metering functions can indicate either positive or negative values. For example, assume the CHANNEL SELECT switch is set on number 10, and that this number selects the measurement of transmitter frequency. The reading indicated by the light-emitting diode (LED) display in the VALUE window may be either above or below the authorized frequency. Thus, the reading is preceded by the proper sign (− or +) to indicate whether the "raise" or the "lower" command signal needs to be operated.

1-7. DIGITIZED AUDIO AND VIDEO

The conversion of aural, visual, and other analog information to digital form is a new and intensely fascinating field. Digital signals can be conveniently "stored" in a memory (Chapter 6). This infor-

mation can then be "read out" on any convenient time base required. You cannot distort a binary bit. You may completely change its position, or even lose it, but you will not distort it. Either it is there or it is not there.

Fig. 1-9 is a block diagram of a typical in-plant application for a television broadcast studio. It makes no difference to the digital system whether the input is from a quad-head or helical-scan vtr, a laser scanner, nonsynchronous network or remote sources, etc. The output signal is in time-base-corrected analog form, is raster- and color-synchronous with the local sync generator, and, in special cases when desired, is changed in format, size, or other characteristics. Processing in digital form is virtually limitless.

Analog information is converted to digital form by sampling the signal and converting each instantaneous sampled level to a corresponding binary number. This process is termed *pulse-code modulation*, or pcm. After processing for whatever functions are to be performed, the binary signals are reconverted to analog form for distribution in the conventional system.

In the example of Fig. 1-9, the system clock operates at three or four times the color subcarrier rate. This provides the reference sampling and timing frequencies.

The analog video signal is fed through a low-pass filter (lpf) to remove noise and harmonics above 4.2 MHz. The signal is then sampled at the clocked rate, which results in a pulse amplitude-modulated (pam) signal. Each sampled level is "held" sufficiently long for the analog-digital (a/d) converter to generate the binary

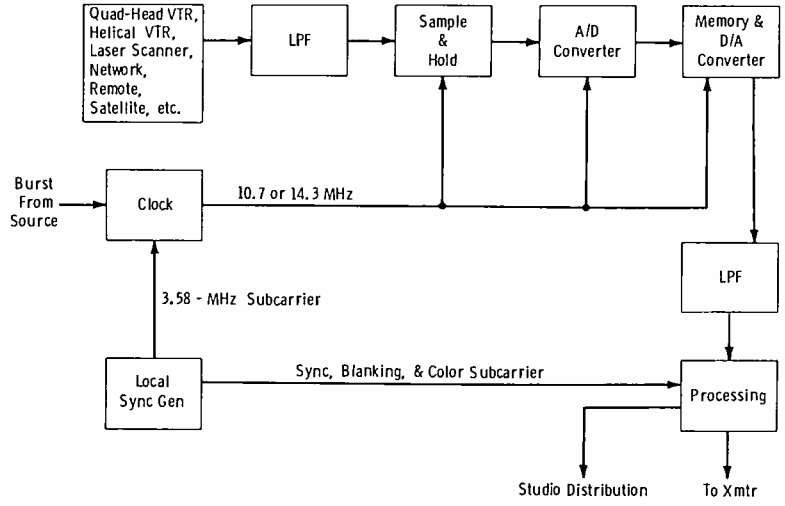


Fig. 1-9. Simplified block diagram of in-plant digital function.

group (byte) associated with the corresponding level. The signal is now in pcm binary form, and it is stored in the memory bank.

The system clock, which is derived from the local sync generator, reads the information from the memory. A digital to analog (d/a) converter feeds the analog information to another low-pass filter to remove switching and pulse transients over 4.2 MHz. The analog signal is now locked element by element to the local generator and is processed for distribution.

A few examples of such applications in television broadcasting are: tape, laser, or other recorders; time-base correctors; image enhancers; automatic synchronizers and timers; standards conversion; video compressors; special effects; all-electronic slide (storage) projectors, etc.

Present-day applications in television use a sampling rate of three times the color subcarrier frequency, or 10.7 MHz. An 8-bit byte is used for each sampled level, providing 256 possible levels to be recognized. Thus, the transmission rate is:

$$(10.7\ \text{MHz})(8) = 85.6\ \text{megabits/second}$$

When parity bits are used, the rate comes close to 100 megabits/second.

If 4 times the color subcarrier is used:

$$(4)(3.58\ \text{MHz}) = 14.3\text{-MHz sampling rate}$$

The transmission rate is:

$$(14.3\ \text{MHz})(8) = 114\ \text{megabits/second.}$$

Transmission rates of this (and greater) magnitude are quite feasible in closed-circuit applications today. For long-distance transmission, such as by rf carrier (lower s/n ratio), bandwidth-reduction schemes are used.

EXERCISES

There is a group of exercises at the end of each chapter; the solutions are given in Appendix B. Try to respond to each question on your own without looking at the answer. If you cannot answer a question, review the chapter carefully and try again. In a few cases, exercises cover an addition to the chapter material, but an addition that is inclusive within the framework of the subject covered. This checks your ability to apply knowledge.

Q1-1. Define the following terms:

(A) bit.
(B) byte.
(C) character.
(D) word.
(E) baud.

Q1-2. Assume a pulse train consisting of a start pulse and five information pulses, all of which have a 1-ms duration. A stop pulse of 4-ms duration is used. Find the:
(A) transmission speed.
(B) rate of information flow.

Q1-3. Assume the stop pulse in the pulse train of Q1-2 is reduced to 2-ms duration. Find the:
(A) transmission speed.
(B) rate of information flow.

Q1-4. What is the total capacity requirement (in bits/s) for the interconnecting link in Q1-3?

Q1-5. Give the basic mathematical relationship for converting (A) hartleys to bits/second and (B) bits/second to hartleys.

Q1-6. If you have a land line with a 5000-Hz bandwidth and an s/n ratio of 50 dB, what is the maximum capacity in bits/second for a baseband signal? (Use common logarithms to get the answer in hartleys; then convert to bits/second).

Q1-7. If each character is made up of 8-bit bytes, how many characters/second would the link of Q1-6 transmit?

Q1-8. If it is desired to maintain the same capacity as that obtained in Q1-6, but the s/n ratio is only 30 dB, what bandwidth will be required? (Hint: Convert bits/second to hartleys and use common logarithms.)

Q1-9. You have a 100-megabit-per-second signal (baseband). What is the required bandwidth for the interconnecting link, assuming the s/n ratio is 50 dB?

Q1-10. In digital applications, is there any use for a link having a noise level the same as the signal level? Assume the available bandwidth is 2500 hertz.

CHAPTER 2

Binary Math

The binary number system provides the simplest form of mathematics in existence. Each element (bit) can be only a 0 or a 1. A pulse represents 1; no pulse represents 0.

2-1. THINKING BINARY

You have heard the mandate "think metric." Let's see what this really means.

Remember the old saying, "An ounce of prevention is worth a pound of cure"? How popular would this saying become if it were in metric form? Converted to metric quantities, the saying becomes, "28.35 grams of prevention is worth 0.4536 kilogram of cure." Certainly no popular saying there! But if you were really "thinking metric," you would simply say, "A gram of prevention is worth a kilogram of cure." That's the point!

Learn to "think binary." This does not mean that we should forget decimals. Rather, we must learn to interface the real world of decimals, letters, and symbols with computer-type machines that use equivalent binary numbers as a language.

When you have completed this chapter, you should continually practice the interfacing of binary numbers with decimal numbers and vice-versa. You will find that the rules become mentally automatic in a short time.

2-2. THE BISTABLE FLIP-FLOP

The bistable flip-flop is the foundation of logic and binary structure. It uses either a positive or a negative trigger (never both) to change its state from off to on, or on to off. It is "stable" in either mode, and it does not change state between triggers.

You know that a flip-flop (or any multivibrator) has two output pulses of opposite polarity. Do not worry about the mechanics at this time. The binary system uses only zeros and ones. A given output is either a 0 (low) or a 1 (high). The outputs reverse in polarity for a change in state.

In Fig. 2-1, note that the flip-flops are sensitive to negative-going edges only. In practice, the "triggering" can be either positive-edge or negative-edge.

Just put the waveforms down as in Fig. 2-1 and analyze them as follows:

Waveform A: The input pulses (waveform A) form a series of zeros and ones. However, since the flip-flops respond only to negative-going transitions, only those transitions that occur at t1, t2, etc., are significant in this example. At time t0, all "1" outputs are 0. We are going to "count," or "accumulate," only the "1" outputs. Thus, the series of flip-flops and associated "storage" can be termed a *counter* or an *accumulator;* either term is correct.

Waveform B: Waveform B is the "1" output of FF1. Note that the first negative-going transition of the input waveform (A) occurs at t1, the next negative-going transition is at t2, and so forth. There-

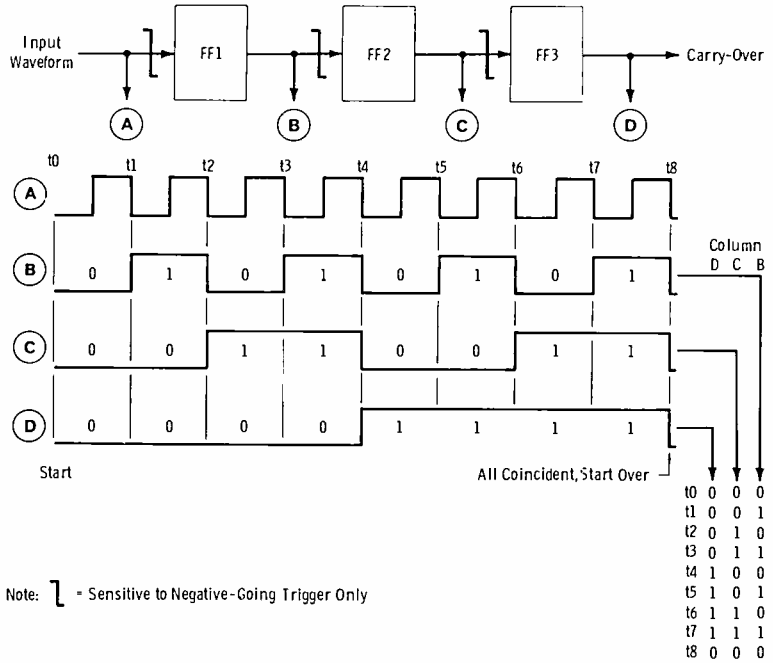


Fig. 2-1. Basic binary formation.

fore, waveform B changes level at each of these times. Write this in column B as a series of alternate 0's and 1's.

Waveform C: The first negative-going transition of waveform B is at t2. This turns FF2 on. The next negative-going transition of waveform B is at t4. This turns FF2 off. Waveform C thus has a series of two 0's and two 1's; put them in column C.

Waveform D: The first negative-going transition of waveform C is at t4. This turns FF3 on (waveform D). The next negative-going transition of waveform C is at t8. It turns FF3 off. This makes a series of four 0's and four 1's; write them down in column D. Note that at this point all waveforms are coincident (0, or low) and the process starts over. You have gone as far as possible with three digits in the binary system.

When you read binary notation, read it only as zeros and ones. Thus, at t0 (Fig. 2-1), read "zero-zero-zero." At t4, do not read "one hundred," but "one-zero-zero."

NOTE: A "set-reset" flip-flop as used in practical circuits is covered in Chapter 4.

Now redraw the output waveforms of Fig. 2-1 as shown in Fig. 2-2. The 1 state of each flip-flop is the shaded area. At time t7, you have "accumulated," or "counted," seven input pulses to get the equivalent of decimal 7. Look at it this way:

The waveforms are in either a 1 state or a 0 state, regardless of pulse duration. For each input pulse, an output pulse is added to one of the waveforms in Fig. 2-2. At t0 you have no pulses (000). At t1 you have one pulse (in waveform B), or 001 = 1. At t2 you have accumulated two 1's, one in waveform B and one in waveform C, for a binary count of 010. Since this 1 is in the 2^1 column, it is equivalent to decimal 2. At this point, you have counted to 2 one time, or 2^1.

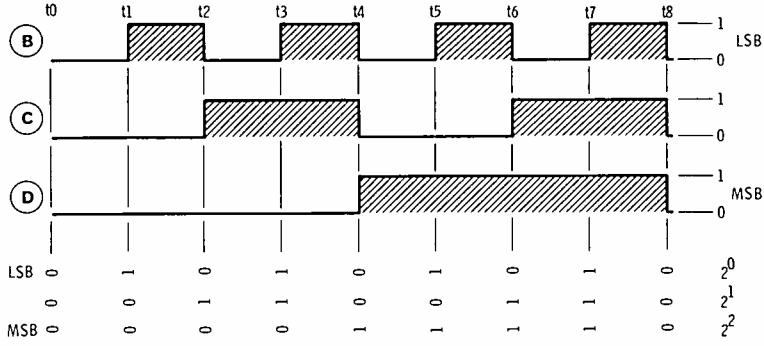


Fig. 2-2. Counting to decimal 7 with three bits.

Continue this reasoning on to t7. At this time, you have four 1's in waveform B, two 1s' in waveform C, and one 1 in waveform D, for a total of 7.

Each flip-flop has divided by 2, which is its basic function. So, you have a single 1 state in waveform D for eight input pulses, or a divide-by-eight divider. The chain of flip-flops is called a *divider, accumulator,* or *counter,* depending on the particular application in which the circuit is used.

2-3. POWERS OF NUMBERS (EXPONENTS)

In our conventional decimal system, $10^1 = 10$ and $10^{-1} = 0.1$. So, 10^0 must lie between 10^1 and 10^{-1} to represent 1. It is a fundamental rule of mathematics that *any* number raised to the zero power is equal to 1. So $2^1 = 2$, $2^{-1} = 0.5$, and $2^0 = 1$.

Table 2-1 reviews the decimal (base-10) system for four digits. The 10^0 column contains units of ones, the 10^1 column contains units of tens, the 10^2 units of hundreds, and the 10^3 units of thousands. Thus, the number $4826 = 4000 + 800 + 20 + 6$. The power is normally termed the *positional weight* of the number. Thus, a 6 in the 10^3 column would represent 6000, but a 6 in the 10^0 column would represent only 6. In Table 2-1, the least significant digit (LSD) is in the 10^0 column, and the most significant digit (MSD) is in the 10^3 column.

Table 2-2 shows the binary system for four bits. Here, instead of ascending powers of ten, we have ascending powers of two. Only zeros and ones are used. A 1 in the 2^0 column (least significant bit, or LSB) is equal to 1. A 1 in the 2^1 column is equal to 2. A 1 in the 2^2 column is equal to 4. A 1 in the 2^3 column (MSB for a 4-bit system) is equal to 8.

To convert any 4-bit binary number to its equivalent decimal value, count from right to left and give each 1 its equivalent decimal value from the "weight" of its given position. For example, bi-

Table 2-1. Decimal System for Four Digits

Positional Weight	10^3	10^2	10^1	10^0	
Equivalent	1000	100	10	1	
	4	0	0	0	
		8	0	0	
4826 =			2	0	$= 4000 + 800 + 20 + 6$
				6	
	= 4 ↑ MSD	8	2	6 ↑ LSD	

Table 2-2. Binary Table for Four Bits

Binary Weight	2^3	2^2	2^1	2^0	
Decimal Equivalent	8	4	2	1	Decimal Count
	0	0	0	0	0
	0	0	0	1	1
	0	0	1	0	2
	0	0	1	1	3
	0	1	0	0	4
	0	1	0	1	5
	0	1	1	0	6
	0	1	1	1	7
	1	0	0	0	8
	1	0	0	1	9
	1	0	1	0	10
	1	0	1	1	11
	1	1	0	0	12
	1	1	0	1	13
	1	1	1	0	14
	1	1	1	1	15
	MSB			LSB	

nary 0110 is simply $2 + 4 = 6$. Binary 1001 is simply $1 + 8 = 9$. Binary $1111 = 1 + 2 + 4 + 8 = 15$.

Going Beyond Four Bits

Note from Fig. 2-1 that the output of FF3 is termed a *carry-over*. This could feed another flip-flop whose output would then be $2^3 = 8$ in decimal form. If added to the waveforms of Fig. 2-1, its output would have a series of 8 zeros and 8 ones. The 1's would start at t8, or the negative-going transition of waveform D. There would then be four bits of information, not counting the input waveform. Table 2-3 lists the positive powers of 2 for values up to decimal 4096. Note that each additional power (weight) doubles the decimal equivalent value. Thus, Tables 2-1 and 2-2 can be expanded to as many powers as required for a particular application. For example: binary $100101 = 32 + 4 + 1 = 37$. Binary $1100101 = 64 + 32 + 4 + 1 = 101$. Simply count the spaces to the left and give each bit the appropriate decimal equivalent; then add. In counting from right to left, just say 1-2-4-8-16-32-64-128-256-etc.

The Modulus

The odometer (mileage indicator) in an automobile normally has a decimal readout capacity of 99,999.9 miles. When you travel an-

Table 2-3. Positive Powers of 2

Power of 2	Decimal Equivalent
2^0	1
2^1	2
2^2	4
2^3	8
2^4	16
2^5	32
2^6	64
2^7	128
2^8	256
2^9	512
2^{10}	1024
2^{11}	2048
2^{12}	4096

other 1/10 mile beyond that, the odometer "resets," or "clears," to 00,000.0, which actually represents 100,000 miles. Therefore, for this device the *modulus* is 100,000. The modulus of any counter is the fixed capacity of the counter plus one increment of the least significant digit (LSD).

Note from Table 2-2 that the 4-bit binary counts from 0 to 15. This is 16 "levels," counting zero, and is the modulus ($2^4 = 16$). Another way to look at this is to realize that the maximum decimal count is the modulus minus 1.

2-4. ADDING BINARY ZEROS AND ONES

The preceding discussion may be entirely logical to flip-flops, but is it logical to you? You cannot draw numerous waveforms to handle binary arithmetic. You will need to work out your arithmetic on paper.

Start by writing down four 0's:

0000 (make this equal to 0)
+ 1 (add 1)
0001 (this equals 1 in the decimal system)

Now remember that decimal 9 is the highest number used in base-10 arithmetic. Thus, when you reach 9, the addition of 1 changes 9 to 0, and a 1 is "carried" to the tens column.

In the binary system, the highest number used is 1. When 1 is added to 1 in the binary system, the sum is 0, and a 1 is carried to the next column to the left. If there is already a 1 in that column, the sum is again 0 and the 1 carries to the left again, etc.

Adding 1 to 0001:

$$\begin{array}{r} 0001 \\ +\ 1 \\ \hline 0010 \end{array}$$

This equals 2 in the decimal system because you have accumulated two 1's, and the 1 is the second space to the left, which is in the 2 column (Table 2-2).

Now add another 1:

$$\begin{array}{l} 0010 \\ \underline{0001} \\ 0011 \end{array}$$ (This is $1 + 2 = 3$ in the decimal system)

Add another 1:

$$\begin{array}{r} 0011 \\ +\ 1 \\ \hline 0100 \end{array}$$

This is 4 in the decimal system, since the 1 is now in the third space to the left ($2^2 = 4$, as shown in Table 2-2).

When you continue this process to 1111, you will have developed the 4-bit binary table of Table 2-2.

For practice, add 0111 and 0110 as in Fig. 2-3. Row 1 contains the augend and row 2 the addend. Row 3 shows the addition of 0 and $1 = 1$ with no carry. Row 4 shows the sum of $1 + 1$, giving 0 and carry 1. In row 5 column C, you already have two 1's with a previous carry of 1. In this case, you must put down your previous carry as shown; then you still carry 1 to the next column. In row 6 column D, you simply record the carried 1. The total, 1101, is in row 7.

Check this addition in decimal form. You know that binary 0111 is equal to decimal 7 ($1 + 2 + 4 = 7$). Also, $0110 = 6$. So $7 + 6 = 13$, and binary $1101 = 13$.

Table 2-4 reviews binary addition. We will refer back to this as we progress.

You have already noted that the digits to the left of the binary point are coefficients of increasing *positive* powers of 2, with 2^0 adjacent to the binary point. The digits to the right of the point are

Row	Column D C B A	
1	0 1 1 1	Augend
2	+ 0 1 1 0	Addend
3	1	No Carry
4	0	Carry 1
5	1	1 Carried and Carry 1
6	1	1 Carried
7	= 1 1 0 1	Binary Total

Fig. 2-3. Addition of binary numbers 0111 and 0110.

Table 2-4. Rules for Binary Addition

BINARY A + B + PREV CARRY =	SUM	NEW CARRY
0 + 0 + 0 =	0	0
0 + 0 + 1 =	1	0
0 + 1 + 0 =	1	0
0 + 1 + 1 =	0	1
1 + 0 + 0 =	1	0
1 + 0 + 1 =	0	1
1 + 1 + 1 =	1	1

coefficients of increasing *negative* powers of 2, with 2^{-1} adjacent to the point.

See Table 2-5 and note that a 1 immediately to the right of the decimal point would be equal to 1/2, or 0.5, whereas a 1 in the fourth place to the right would be equal to 1/16, or 0.0625. Thus, binary 0.1 = decimal 1/2 or 0.5, and binary 0.0001 = decimal 1/16 or 0.0625. For example, the decimal equivalent of binary 0101.01 is:

Binary	*Decimal*
0×2^3	$= 0$
1×2^2	$= 4$
0×2	$= 0$
1×2^0	$= 1$
0×2^{-1}	$= 0$
1×2^{-2}	$= 0.25$
0101.01	$= 5.25$

In practice, of course, you simply disregard all zeros and note the binary weight of the 1 for each position. When you become familiar with binaries, you can look at 0101.01 and say: 4 + 1 + 0.25 = 5.25. In a 4-bit binary number, you simply have (from left to right) 8-4-2-1 plus any fractional quantity.

2-5. SUBTRACTING BINARY ZEROS AND ONES

On paper, binary subtraction is no different from decimal subtraction; the same "borrowing" rules apply. For example:

Table 2-5. Binary Numbers With Fractions

Binary Weight	2^3	2^2	2^1	2^0	POINT	2^{-1}	2^{-2}	2^{-3}	2^{-4}
Decimal Equivalent	8	4	2	1	.	1/2 (0.5)	1/4 (0.25)	1/8 (0.125)	1/16 0.0625

$$\begin{array}{r} 1111 \text{ (Decimal 15)} \\ -\ \underline{0110} \text{ (Decimal 6)} \\ = 1001 \text{ (Decimal 9)} \end{array}$$

In this case, binary subtraction is simpler than the decimal form, since 15 − 6 requires "borrowing" the 1.

Now suppose you have:

$$\begin{array}{r} 44.00 \\ -\ \underline{10.25} \\ = 33.75 \end{array}$$

Observe here that you borrowed twice until a number existed (third digit from right) to borrow from. The computation 44 − 10.25 in binary form is:

$$\begin{array}{r} 101100.00 \text{ (Decimal 44)} \\ -\ \underline{001010.01} \text{ (Decimal 10.25)} \\ = 100001.11 \text{ (Decimal 33.75)} \end{array}$$

Making the first digit the one to the extreme right, the procedure is as follows:

1st digit: 0 − 1 is a difference of 1 and borrow 1.
2nd digit: No 1 exists to borrow yet, so 0 in the top row becomes 1, and 1 − 0 = 1 and borrow 1 carried over.
3rd digit: Still no 1 to borrow from, so 0 in the top row becomes 1, and 1 − 0 = 1 and borrow 1 carried over.
4th digit: Still no 1 to borrow from, so 0 in the top row becomes 1, and 1 − 1 = 0 with borrow 1 carried over.
5th digit: Now a 1 exists to borrow from, so the 1 becomes a 0, and 0 − 0 = 0. No carry.

The rest of the procedure is self-explanatory.

Another example:

$$\begin{array}{r} 0110 \text{ (Decimal 6)} \\ -\ \underline{0011} \text{ (Decimal 3)} \\ = 0011 \text{ (Decimal 3)} \end{array}$$

1st digit: 0 − 1 = difference of 1 and borrow 1.
2nd digit: 1 is borrowed from the top row (becomes 0), and 0 − 1 is a difference of 1 and borrow 1.
3rd digit: 1 is borrowed from the top row (becomes 0), and 0 − 0 = 0.
4th digit: 0 − 0 = 0.

Table 2-6 lists the rules for binary subtraction.

When we progress to the complement (next section), you will find binary subtraction much simplified.

Table 2-6. Rules for Binary Subtraction

A − B − Previous Borrow	= Difference	New Borrow
0 − 0 − 0	= 0	0
0 − 0 − 1	= 1	1
0 − 1 − 0	= 1	1
0 − 1 − 1	= 0	1
1 − 0 − 0	= 1	0
1 − 0 − 1	= 0	0
1 − 1 − 0	= 0	0
1 − 1 − 1	= 1	1

2-6. HANDLING POSITIVE AND NEGATIVE BINARIES

There are positive and negative values in binary notation just as in base-10 arithmetic. It is time now to consider the binary scheme for manipulating either polarity in calculations. You will find binary subtraction by complements much simpler than the straight arithmetical method described above.

The Complement

A *complement* is that quantity or amount which, when added to a given quantity, completes a whole. For example, in Fig. 2-4, if you want to complete an angle of 90°, arc BD is the complement of arc AB, and angle BCD is the complement of angle ACB. Another way of saying this is that, for an angle of 90° as a whole, the complement of 30° is $90 - 30 = 60°$.

The complement of $460 = 1000 - 460 = 540$. The complement of 25 is $100 - 25 = 75$. Note that in each case the arithmetical complement is actually the *difference* obtained by subtracting the number from the next higher power of the base. When you complement a number, the number becomes a *negative* value, and the complement is a *positive* number. Synonyms for "complement" are: inversion, reversal, opposite, reverse, inverse, and converse.

In conventional arithmetic, suppose you subtract 235 from 485:

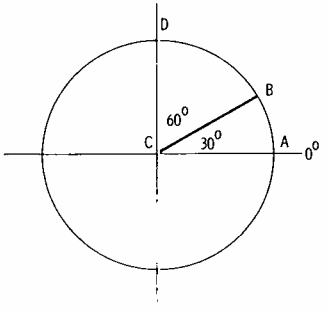


Fig. 2-4. Complement of an angle.

485 (Minuend)
− 235 (Subtrahend)
= 250 (Remainder)

You can do the same thing by adding rather than subtracting if you complement the subtrahend and continue as in normal addition:

485 (Minuend)
+ 765 (Complement of subtrahend)
*1*250 (Remainder with *1* overflow)

In complementary arithmetic, the most significant digit (*1* in the above example) is not a part of the answer numerically, and the answer (250) remains.

Let us see what happens if the result is a negative number rather than a positive number. Suppose you have 235 − 485. Normally, you simply subtract the smaller number from the larger number and affix the sign of the larger number to the answer. Hence, −485 + 235 = −250.

In complementary arithmetic:

235 (Minuend)
+ 515 (Complement of a negative number is positive)
= 750 (Complement of 250)

Then recomplementing 750 = −250 (answer). The complement of a complement is the original number.

Simply note from this example that when a resultant is a negative number, it will be indicated by the appearance of the complement form in the answer. The answer is then recomplemented to get the correct negative value.

Binary Complements

Complements in binaries are much simpler than conventional complementary arithmetic. To complement a binary, simply invert the number (change all zeros to ones and all ones to zeros), as you would do in a single electronic polarity-inverter stage. This is called the *ones complement*. Then add 1 for the modulus of the counter (called the *twos complement*). For example, to complement binary 0010 (decimal 2):

1101 (Inverted 0010—ones complement)
+ 1 (Add 1)
1110 (Twos complement of 0010)

Remember that the modulus for a 4-bit number is 15 + 1 = 16 ($2^4 = 16$). Thus, the complement of 2 in this system is 16 − 2 = 14 = binary 1110 as derived above.

Now consider the addition of 0110 and −0100 (decimal 6 − 4). Simply complement the negative number (which gives you a positive number), and then add the result to the positive number:

```
  1011  (Inverted 0100—ones complement)
+    1  (Add 1)
  1100  (Complement of 0100—twos complement)

  0110
+ 1100
 10010  (Answer)
```

Now discard the overflow digit (MSD), and replace it with either a plus or a minus sign. If the MSD is 1, the sign is positive. If the MSD is zero, the sign is negative. In the above example, the MSD is 1, so your answer is +0010, or decimal 2. This satisfies the condition that 6 − 4 = 2.

When a negative number is added to a smaller positive number, the overflow (MSD) is always 0, and the answer is negative. When the result is negative, the number must be recomplemented to obtain the correct solution. For example, add −1001 (decimal −9) to 0010 (decimal 2). First find the complement of 1001:

```
  0110  (Inverted—ones complement)
+    1  (Add 1)
  0111  (Complemented—twos complement)
```

Now add this to 0010:

```
  0111
+ 0010
 01001
```

Since the MSD is a 0, we discard it and write a negative sign to get −1001. Now to recomplement:

```
  0110  (Inverted)
+    1  (Add 1)
  0111  (Negative answer)
```

Thus, the answer is −0111. This is decimal −7, which is the result of adding −9 to +2.

Binary subtraction by complements is exactly the same as addition of positive and negative binary numbers. For example, subtract 0011 (decimal 3) from 1111 (decimal 15). First find the complement of 0011:

```
  1100
+    1
  1101
```

Then add:

$$\begin{array}{r} 1111 \\ \underline{1101} \\ 11100 = +1100 = \text{Decimal } 12 \end{array}$$

2-7. BINARY MULTIPLICATION

Binary multiplication is identical to decimal (base-10) multiplication. However, in the decimal system, you must have a multiplication table that includes the numbers from 1 to 9 multiplied by every number from 1 to 9. In the binary system, the only digits are zero and one; therefore; a highly simplified multiplication table results.

For example, $7 \times 3 = 21$. Here you said to yourself "7-14-21." In binary notation, 7×3 is:

$$\begin{array}{r} 0111 \\ \times\ \underline{0011} \\ 111 \\ \underline{111} \\ \text{TOTAL} \end{array}$$

Note that all you have are the 1's in their proper weight columns. Since $0 \times 1 = 0$, you have only two rows in this example to add for the total:

$$\begin{array}{r} 111 \\ \underline{1110} \\ 1 \quad (0 + 1 = 1) \\ 0 \quad (1 + 1 = 0 \text{ and carry } 1) \\ 1 \quad (1 + 1 + \text{carry } 1 = 1 \text{ and new carry } 1) \\ 0 \quad (1 + \text{carry } 1 = 0 \text{ and carry } 1) \\ \underline{1} \quad (1 \text{ carried}) \\ = 10101 = 16 + 4 + 1 = 21 \end{array}$$

We have just reviewed the binary addition technique of Fig. 2-3 and Table 2-4 for convenience. Remember that binary multiplication is identical to base-10 multiplication, but you must use the proper binary *addition* technique.

You can observe an interesting and useful point at this time: Binary multiplication is extremely simple, and binary addition takes more practice than multiplication. Binary multiplication requires no complicated tables at all since it involves only 0's and 1's. When you have had considerable practice in binary addition (and sub-

traction), you will feel entirely at home in its application. This is very important because you will soon discover that a computer actually multiplies by over-and-over addition, and it divides by over-and-over subtraction.

Another important correlation between the decimal and binary systems is as follows: In the decimal (base-10) system, if (for example) 12 is shifted left one position (120), you have multiplied once by 10. If 12 is shifted left two places (1200), you have multiplied twice by 10. In the binary (base-2) system, if 1100 (decimal 12) is shifted left one position (11000, decimal 24), you have multiplied once by 2. If 1100 is shifted left two places (110000, decimal 48), you have multiplied twice by 2.

This should emphasize in your thinking that a one-space shift in the binary system is simply a change of one power of 2 rather than a change of one power of 10 as in the conventional decimal system.

As another example of multiplication, take 12 × 13:

$$\begin{array}{r} 12 \\ \times\ 13 \\ \hline 36 \\ 12 \\ \hline = 156 \end{array}$$

In binary form, 12 × 13 is:

$$\begin{array}{r} 1100 \\ \times\ 1101 \\ \hline 1100 \\ 0000 \\ 1100 \\ 1100 \\ \hline = 10011100 \end{array}$$

(Note that this row is not necessary.)

(See Fig. 2-5 for conversion to decimal.)

People do binary multiplication on paper as just described, but computers multiply by over-and-over addition. A 5-MHz flip-flop performs five million such operations per second.

Weight	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Value	128	64	32	16	8	4	2	1
Binary:	1	0	0	1	1	1	0	0
*	128			+16	+8	+4		

* 156 (Answer)

Fig. 2-5. Conversion of binary 10011100 to decimal value.

A calculator, whether the "old-fashioned" mechanical type or the latest electronic digital type, makes use of two registers, the *accumulator* and the *multiplier*. The accumulator starts with all zeros, and the multiplier keeps count of the number of additions performed.

Let us see how a base-10 calculator finds the product of 12×13 with over-and-over addition. Refer to Fig. 2-6, and observe the following step-by-step operations:

	Accumulator	Multiplier
1.	000	13
2.	012	
3.	012	12
4.	012	
5.	024	11
6.	012	
7.	036	10
8.	120	
9.	156	0 (Answer)

Fig. 2-6. Base-10 multiplication of 12 × 13.

In Step 1, we record zeros in the accumulator column and the multiplier (13) in the multiplier column. In Step 2, we place the multiplicand in the accumulator. In Step 3, we have the first addition ($000 + 012 = 012$). Since we have performed one addition, we reduce the multiplier by 1 (13 reduced to 12). In Step 4, we add the multiplicand again. Step 5 is the new subtotal, and the multiplier is again reduced by 1. In Step 6, we add again; the new subtotal is in step 7, where the multiplier has now been reduced to 10. When the first-order digit of the multiplier is reduced to 0 as it is

	Accumulator	Multiplier	Procedure
1	00000000	1101	
2	1100		Add.
3	1100	1100	Added and multiplier reduced by one.
4	1100		Multiplicand shifted two places left.
5	1111	1000	Added. Cancels third digit in multiplier.
6	1100		Multiplicand shifted one space left.
7	10011100	0000	This is the answer.

= 128 + 16 + 8 + 4 = 156 Decimal

Fig. 2-7. Binary multiplication of 12 × 13.

in Step 7, in the next step (Step 8), the multiplicand is moved one place to the left as shown. Now the multiplicand will be added 10 times in each step. Thus, in Step 9 the multiplier has been reduced to 0, and the total in the accumulator is the answer.

The binary method of obtaining the product 12×13 is shown by Fig. 2-7. Here are the step-by-step operations:

In Step 1, we record 0's in the accumulator and the multiplier in the right-hand column. In Step 2, the multiplicand is recorded in the accumulator. Step 3 is the first addition, which reduces the multiplier by 1 (1101 reduced to 1100). Now note that the *last two* digits of 1100 are 0's, so in the next step (Step 4) the multiplicand is shifted *two* places to the left as shown. Step 5 is the new total, and the third digit in the multiplier is cancelled. Thus, the multiplicand is moved one place to the left again for Step 6. The new total in Step 7 is the answer, since the multiplier has been reduced to 0.

Let us look at a very simple example, $8 \times 8 = 64$ in binary form. In binary notation, $8 = 1000$. The entries in the accumulator register and the multiplier register are:

	Accumulator	*Multiplier*
	00000000	1000
	1000- - -	
Total =	1000000	0000

Notice that the three 0's in the multiplier shift the multiplicand three places to the left in the accumulator to get to the next digit in the multiplier. Thus, in this example the very first addition reduces the multiplier to 0 (cancels the fourth digit) to obtain the answer 1000000 (64 in decimal form). At this point, you should be able to see immediately that since the 1 is in the seventh space to the left, it is equivalent to decimal 64. Although it is not necessary to memorize complicated multiplication tables, it *is* convenient to memorize the positional weights of powers of 2, such as in Fig. 2-5 and Table 2-3.

2-8. BINARY DIVISION

Binary division on paper is identical to base-10 division. For example, divide 15 by 5:

Base-10 Division	*Binary Division*
3	0011
5/15	101/1111
15	101
00	101
	101
	000

As another example, divide 16.75 by 4 in base-10 and binary forms:

Base-10 Division

```
     4.1875
  4 ) 16.7500
      16
      ---
       7
       4
      ---
       35
       32
      ---
        30
        28
       ---
         20
         20
        ---
          00
```

Binary Division

```
      100.0011
  100 ) 10000.1100
        100
        ---
        00000 110
              100
              ---
               100
               100
               ---
                 000
```

As a check, the decimal equivalent of the binary answer, 100.0011, is:

$$4 + 0.125 + 0.0625 = 4.1875$$

If this is giving you trouble, review Table 2-5.

Division can be accomplished in either the decimal system or the binary system by over-and-over subtraction. For example, 18 divided by 6 is:

```
  18
- 6
  --
  12   First subtraction
- 6
  --
   6   Second subtraction
- 6
  --
   0   Third subtraction
```

The number of subtractions is three, so $18/6 = 3$.

In binary form, $18 - 6$ is:

```
  10010
-00110
  -----
  01100   First subtraction
-00110
  -----
  00110   Second subtraction
-00110
  -----
  00000   Third subtraction
```

Here again, the number of subtractions is three, which in binary form is 0011. On paper:

```
     00011.0
110/10010.0
    110
    00110
    00110
    00000
```

Remember that the binary division technique is identical to base-10 methods, but you must use the binary subtraction technique as covered in Section 2-5.

2-9. CONVERTING AND SIMPLIFYING NUMBER SYSTEMS

Conventional decimal arithmetic normally does not have the radix (10) indicated in the number. Thus, 1024 (read "one thousand twenty-four") is usually written as such rather than 1024_{10}. The radix, or base, 10 is assumed.

When numbers of more than one base are being considered, the radix must be indicated. For example:

10_{10} = Decimal 10
10_8 = Octal 10 = Decimal 8
10_2 = Binary 10 = Decimal 2

Any number of a given base can be converted to a different number of another base. The "different" answer is actually an *equivalent number* if the base is specified. For example, 13_{10} in decimal form = 1101_2 in binary form. This section will explain the techniques involved in converting and simplifying the most common numbering systems.

Decimal to Binary

You have discovered from previous study how easy it is to convert binary numbers to decimal numbers. You must also be able to convert decimal numbers to binary numbers.

In Fig. 2-8, you can see that binary 10001100100.11 is equal to $1024 + 64 + 32 + 4 + 0.5 + 0.25 = 1124.75$. Now let us see how 1124.75 is converted to binary notation.

Step 1 (Fig. 2-8) is to record the number to be converted. Step 2 is to write down the largest whole power of 2 (in decimal form) that can be subtracted from the number in Step 1. This is 1024, so you place a 1 under the decimal value of 1024 in the bottom row of the table in Fig. 2-8. Step 3 is to find the remainder, and Step 4 is to find the largest whole power of 2 that can be subtracted from the 100.75 of Step 3. You place a 1 under 64, and so on through all

Powers of 2	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	.	2^{-1}	2^{-2}
Decimal Value	2048	1024	512	256	128	64	32	16	8	4	2	1	.	0.5	0.25
Binary for 1124.75	0	1	0	0	0	1	1	0	0	1	0	0	.	1	1

STEP		NOTES	OPERATION
1	1124.75	Number To Be Converted	
2	-1024.00	Largest Whole Power of 2 That Can Be Subtracted ------	Place a 1 Under 1024
3	100.75	Remainder	
4	-64.00	Largest Whole Power of 2 That Can Be Subtracted ------	Place a 1 Under 64
5	36.75	Remainder	
6	-32.00	Largest Whole Power of 2 That Can Be Subtracted ------	Place a 1 Under 32
7	4.75	Remainder	
8	-4.00	Largest Whole Power of 2 That Can Be Subtracted ------	Place a 1 Under 4
9	0.75	Remainder	
10	-0.50	Largest Whole Power of 2 That Can Be Subtracted ------	Place a 1 Under 0.5
11	0.25	Remainder	
12	-0.25	Largest Whole Power of 2 That Can Be Subtracted ------	Place a 1 Under 0.25
13	0.00	Remainder	

Fig. 2-8. Steps in converting decimal 1124.75 to binary equivalent.

remaining steps. Then you write a 0 under all values not used in the above procedure. Thus, you find that decimal 1124.75 = 10001100100.11. After you have memorized the powers of 2 and their decimal equivalents, and had a lot of practice (which is required in *any* math), you will be able to write the binary equivalent of a decimal number without actually drawing up a table.

An alternative method of converting decimal numbers to binary form is illustrated in Fig. 2-9. In this example, 54_{10} is converted to binary form by repeatedly dividing by 2 and recording the remainders. The LSB is at the top and the MSB at the bottom, so you read from the bottom up to get the equivalent binary number. Double-check the binary number by noting that 110110 is 32 + 16 + 4 + 2 = 54 decimal.

Again note that you will be able to find the resulting binary number mentally with a little practice. Remember to reverse the result to put the MSB first.

As you know by now, binary notation can become very long in representing decimal values. We are now ready to study various number-conversion techniques that simplify binary numbers.

Pure Binary and BCD

Thus far, we have been concerned with the pure binary code, which uses the exact positional weight (as in Fig. 2-8) of each digit as the weight value. The binary-coded decimal (bcd) code employs four binary *bits* per *character* and has the weight scheme of 8-4-2-1. You already know that four bits can represent a pure binary num-

Decimal Number	÷ Base 2	= Result	Remainder (Binary)	
54	÷ 2	= 27	0	LSB
27	÷ 2	= 13	1	
13	÷ 2	= 6	1	
6	÷ 2	= 3	0	Read Up
3	÷ 2	= 1	1	
1	÷ 2	= 0	1	MSB

= Binary 1 1 0 1 1 0

Fig. 2-9. Conversion of decimal 54 to binary form.

ber from 0 to 15. For bcd, only the ten decimal numbers 0 to 9 are normally employed. To express decimal numbers greater than 9, a separate bit group is used for each digit

For example, 92 in pure binary form = 1011100. In bcd form, 92 is:

$$\underbrace{1001}_{9} \qquad \underbrace{0010}_{2}$$

Note that the first group is equivalent to decimal 9 and the second group is equivalent to decimal 2. Each group simply represents a number from 0 to 9. For 920, the pure binary equivalent is 1110011000. In bcd, this number is:

$$\underbrace{1001}_{9} \qquad \underbrace{0010}_{2} \qquad \underbrace{0000}_{0} = 920 \text{ bcd}$$

In one way, this greatly simplifies the conversion of decimal numbers to binary notation. For example:

$$75 = 0111\ 0101$$
$$755 = 0111\ 0101\ 0101$$
$$8755 = 1000\ 0111\ 0101\ 0101$$

It is not necessary to space the groups of four binary numbers in bcd provided the function bcd is specified.

$$01110101 \text{ bcd}$$

In this instance, you mentally mark off the groups of four:

$$\underbrace{0111}_{7}\underbrace{0101}_{5} \text{ bcd} = 75 \text{ bcd}$$

Still, it will be observed that many bits are needed to represent equivalent decimal numbers.

Octal-Coded Binary

The octal numbering system is frequently used in digital systems because it greatly simplifies the handling of the long strings of 0's and 1's required to represent a large equivalent decimal number in binary form. The position weighting is based on powers of 8. The octal system converts readily to binary form, and *vice-versa*, because the base of the octal system (8) is an integral power of 2 ($8 = 2^3$).

Since the octal system has 2^3 as its base, we can arrange any long string of zeros and ones in a binary number into groups of three bits each; this converts the binary number to equivalent octal form. Let us see how simple this is.

In Fig. 2-10, the pure binary word 100010.01 has been arranged in groups of three bits each under the corresponding positional weights 4-2-1 for octal coding. Beginning at the binary point, mark off the binary number into groups of three digits to the left and to the right of the point. Add a 0 where needed to complete a group of three.

When converting from octal to decimal form, it is most convenient to use the octal-to-binary conversion as an intermediate step. This is shown in rows C, D, and E of Fig. 2-10. Then the binary number is converted to decimal form in the normal manner, as shown in rows E and F. Note carefully in row A that the octal weight 4-2-1 applies to all blocks of three, even those to the right of the decimal point. Thus, pure binary 100010.01 = 42.2 octal = 34.25 decimal.

With some practice, all the steps in converting from decimal to octal and from octal to decimal, using pure binary as an intermediate step, can be done mentally.

For example, what is the decimal equivalent of octal 26.1? Just visualize the number as follows:

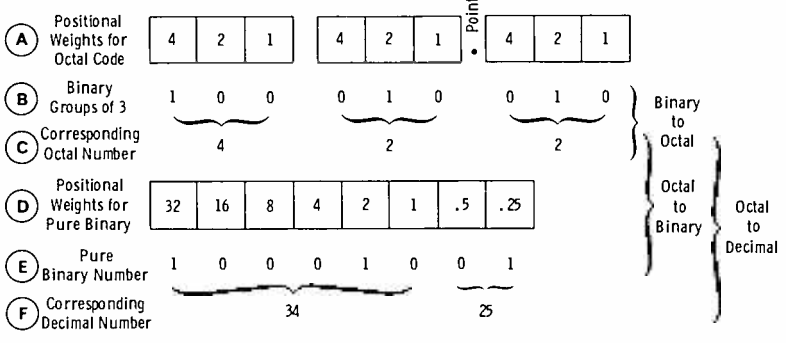


Fig. 2-10. Binary-to-octal and octal-to-decimal conversion.

Octal:	2	6	.	1
	↓	↓		↓
Binary:	010	110	.	001 = pure binary 10110.001

Then convert from binary to decimal form by powers-of-2 weighting and summation: 010110.001 = 16 + 4 + 2 + 0.125 = 22.125 decimal.

For another example, what is the octal equivalent of decimal 568? Using the procedure of Fig. 2-8 or 2-9, decimal 568 = 1000111000 in binary form. Then converting this to octal form:

Binary:	001	000	111	000	
	↓	↓	↓	↓	
Octal:	1	0	7	0	= octal value of 1070

Keep Your Perspective

The binary system, keyed to the decimal and octal systems, is used in many digital devices and in the great majority of computers. Therefore, we have stressed the correlation of the binary notation with decimal and octal values. However, the octal (base-8) system can be represented as increasing powers of 8 just as the decimal system (Table 2-1) is expressed in increasing powers of 10 and the binary system (Table 2-2) is expressed in increasing powers of 2. This is illustrated in Table 2-7. While increasing powers of 8 are not easily handled, it is pertinent to keep the proper perspective in all number systems.

For example, octal 26.1 is equal to $(2 \times 8^1) + (6 \times 8^0) + (1 \times 8^{-1})$. This is $(2 \times 8) + (6 \times 1) + (1 \times 0\ 125) = 16 + 6 + 0.125 = 22.125$ decimal.

2-10. THE FLEXIBLE BINARY

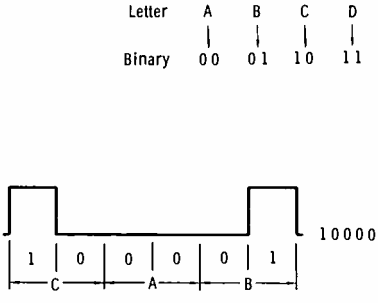
Binary pulses can represent anything, not just numbers. Fig. 2-11A shows the letters A, B, C, and D with arbitrarily assigned binary values. Note that all possible combinations of the two bits are used: 00, 01, 10, and 11. Thus, a 2-bit binary word can define only four characters ($2^2 = 4$). Fig. 2-11B shows that in this example, binary 100001 represents the word *CAB*.

Similarly, as illustrated by Fig. 2-11C, a 2-bit binary can represent any of four possible levels or voltage magnitudes.

To define all the letters of the alphabet (26), you need five bits ($2^5 = 32$). Note that four bits ($2^4 = 16$) would not be enough.

Table 2-7. The Octal System

Octal Weight	8^3	8^2	8^1	8^0	.	8^{-1}
Decimal	512	64	8	1	.	0.125



(*A*) *Letters with arbitrary binary values.*

(*B*) *Example of binary coded word.*

(*C*) *Four levels.*

Fig. 2-11. Binary representations.

EXERCISES

Q2-1. Convert the following binary numbers to their decimal (base-10) equivalents:
(A) 101001.01 (C) 11101010.101
(B) 1011.1111 (D) 0001.0001

Q2-2. Find the binary sum and give the decimal equivalent of:
(A) 1000 + 1001 (C) 0000.1110 + 0001.1110
(B) 1101.1 + 1000.101 (D) 0000.11111 + 0011.111

Q2-3. Find the difference between the following by using two methods, straight arithmetical and complements:
(A) 1110 − 0110
(B) 101001 − 001010

Q2-4. Find the products by over-and-over addition:
(A) 1011 × 1110 (C) 1111 × 1111
(B) 1011 × 11010

Q2-5. Find the following quotients. Remember that a quotient is the result of division—the number of times one quantity is contained in another. Therefore, solve by two different methods, straight arithmetical division and over-and-over subtraction.
(A) 100100/1100
(B) 1000/0010

Q2-6. Convert the following decimal numbers to binary notation:
(A) 124.875 (C) 0.53125
(B) 262 (D) 187.625

Q2-7. Write the following decimal numbers in bcd form:
(A) 864.2 (C) 25.8
(B) 1087 (D) 265

Q2-8. Convert the following binary numbers to octal form:
(A) 1000110100
(B) 111100100.01

Q2-9. Convert the following octal values to decimal form using the binary form as an intermediate step:

(A) 744.2
(B) 1064
(C) 272.5
(D) 31.7

Q2-10. Convert the following octal numbers to decimal form using straight octal conversion (no intermediate step):

(A) 31.1
(B) 1064

Q2-11. Solve by the additive method:

(A) 110101 − 001011
(B) 1111 − 11010
(C) 1110101 − 1111010
(D) (1001 0100) − (0001 0010) This is bcd.
(E) (0001 0010) − (1001 0100) This is bcd.

Q2-12. How many straight binary digits (bits) must be used to "recognize" the following number of characters or levels?

(A) 10
(B) 100
(C) 1000
(D) 10,000
(E) 25,000

CHAPTER 3

Logic

In 1847, a mathematician named George Boole demonstrated that logical statements could be written in simplified symbolic forms and manipulated very much like algebraic symbols. Symbolic logic and Boolean algebra are one and the same thing. Binary logic is the very foundation of all digital technology.

3-1. PREREQUISITE

The only prerequisite to this chapter (other than a knowledge of basic mathematics and electronics) is the following:

In binary logic, symbols such as +, ·, 0, and 1 have nothing to do with numerical algebra. For example, A · B, or simply AB, is *not* "A times B," but rather "A AND B." Similarly, A + B is *not* "A plus B," but rather "A OR B." Letters ABC or XYZ, etc., have a value of either 0 or 1, "low" or "high." And for proper perspective, "high" can be ground (zero volts) if "low" is a negative voltage value.

3-2. LOGIC STATEMENTS

Let us make two statements:

A. "The oscillator output level is normal."

AND ←——connective

B. "The oscillator frequency is in tolerance."

A statement is either true (T) or false (F). A function (f) is the result, or function, of the *combined* statement A AND B (AB). The connective (AND in this case) is also called the *operator.*

So, if you say "the oscillator output level is normal AND the frequency is within tolerance," the function of this combined state-

ment can be true only if *both* A AND B are true. If either or both are false, the combined statement (f) is false. Put down the *truth table* for this as in Fig. 3-1. Your reasoning should be as follows:

Row 1. If A is false and B is false, the combined function (f) is false.
Row 2. If A is false and B is true, the combined function is false.
Row 3. If A is true and B is false, the combined function is false.
Row 4. If A is true and B is true, the combined function is true.

You must become familiar with truth tables. Most component specifications are given in truth-table form. You will troubleshoot components and entire systems by comparing actual performance with the truth table. Practice using the truth table form until you can draw up a truth table to meet any specific requirement. The technique will be emphasized in this chapter.

3-3. THE CONNECTIVE (OPERATOR) AND

In the combined statement of Section 3-2, the connective, or operator, AND was used. Basic logic is founded on either a true or a false condition, with no "gray area" in between. A true or false "ANDed" condition can be very simply represented by a closed or open switch, as shown by Fig. 3-2A. If either A or B is open, there will be no current through R. If A AND B are closed, there will be current through R.

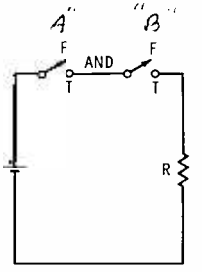
Let a closed switch = true = 1, and let an open switch = false = 0. Then for the statement of Section 3-2, Fig. 3-2B shows the truth table in *binary* notation.

The schematic symbol for an AND circuit is shown by Fig. 3-2C. Whenever you see this symbol, you must recognize the AND function.

Fig. 3-3 is a summary of the AND function. In row 1, switch A must be a two-contact series arrangement. Therefore, the function depends entirely on the "A" mode. The function is read "A AND A = A."

Fig. 3-1. Truth table for AND function.

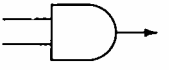
Row	A	B	f = AB
1	F	F	F
2	F	T	F
3	T	F	F
4	T	T	T



(A) *Circuit with switches.*

A	B	f = AB
0	0	0
0	1	0
1	0	0
1	1	1

(B) *Binary truth table.*



(C) *Schematic symbol.*

Fig. 3-2. AND circuit.

In row 2, since the 1 is a closed circuit, the function depends entirely on the "A" mode. The function is read "A AND 1 = A."

In row 3, the 0 is an open circuit. Therefore, the function will always be 0 regardless of the "A" mode. The function is read A AND 0 = 0."

An open or closed switch can be represented by Boolean notation. If A = 1, $\overline{A} = 0$. The bar over the A ($\overline{A}$) represents inverted A. If B = 1, $\overline{B} = 0$. The bar over the B ($\overline{B}$) represents inverted B.

Fig. 3-4 includes the Boolean notation in the AND truth table. When switch A is closed (binary 1), the Boolean notation is A.

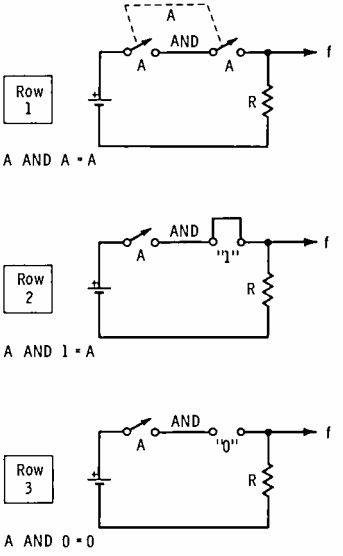


Fig. 3-3. Summary of AND circuit.

Binary		Boolean	f = AB	Comments
A	B			
0	0	$\bar{A}\bar{B}$	0	Reads "A NOT AND B NOT = 0"
0	1	$\bar{A}B$	0	Reads "A NOT AND B = 0"
1	0	$A\bar{B}$	0	Reads "A AND B NOT = 0"
1	1	AB	1	Reads "A AND B = 1"

Fig. 3-4. AND truth table with Boolean notation added.

When switch A is open (binary 0), the Boolean notation is $\bar{A}$ (read as "A NOT"). The same notation prevails for switch B.

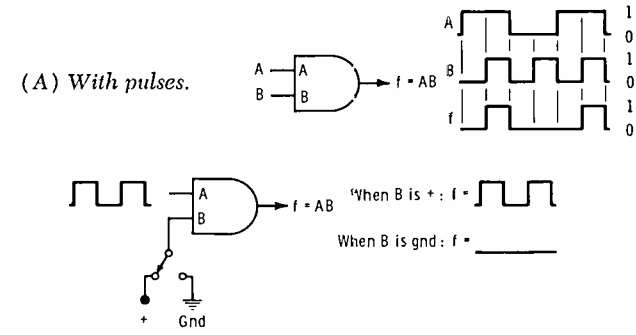
NOTE: $\bar{A}$ can also be read as "NOT A." Similarly, $\bar{B}$ can be read as "NOT B." However, there are times when the "A NOT" and "B NOT" reading is preferable; an example of this will be shown in Section 3-5.

Logic circuitry is normally called a *gate*. Fig. 3-5A shows a pulse application in which pulses of different widths are applied to inputs A and B. An output from the AND gate can occur only when both inputs are high (coincident). Therefore, the input on A is "gated" at times when the input on B is coincidently high.

Fig. 3-5B shows an AND gate where input B is a dc voltage that is switched between plus (high) and ground (low). Thus, the A input is "gated" to the output only when input B is high (positive voltage). In general, you should understand that the logic symbol is used either for a logical operation or a gated operation.

3-4. THE CONNECTIVE OR

Another basic logic function is the OR circuit. There are two types of OR functions:



(A) With pulses.

(B) With dc control.

Fig. 3-5. Operation of an AND gate.

1. The Inclusive OR, either one or both (and/or).
2. The Exclusive OR, either one but not both (either/or but not both).

The Inclusive OR

Let us make two statements:

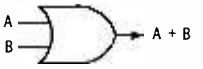
A. The tape frame pulse is present.
OR ←——— connective
B. The edit pulse is present.

This procedure is used in fast lock-up capstan servos in video tape recorders, where the two pulses are of the same frequency and have the same relative timing. A comparison of *either* or *both* of these is made with the reference frame pulse.

Symbols: A OR B = A + B.
f = function of combined statement A + B (A OR B).

If you say "the tape frame pulse is present OR the edit pulse is present," the combined statement will be true if A is true, OR B is true, OR *both* A and B are true (one or both). This is the inclusive OR function. The schematic symbol is shown in Fig. 3-6A, and the Inclusive OR truth table is shown in Fig. 3-6B.

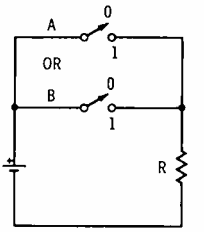
Inclusive OR logic can be very simply represented by open or closed switches as in Fig. 3-6C. If switch A OR switch B is closed, there will be current in R. If both A and B are closed, there will be current in R. If A and B are both open, there will be no current. As before, a closed switch can represent 1, and an open switch can represent 0.



(*A*) *Schematic symbol.*

A	B	f = A + B
0	0	0
0	1	1
1	0	1
1	1	1

(*B*) *Truth table.*



(*C*) *Switch representation.*

Fig. 3-6. Inclusive OR function.

Note carefully that the Inclusive OR circuit can be either or both. The usual definition is that $f = A + B$ The complete definition is actually that $f = A + B + AB$. This reads "f = A OR B or A AND B."

The combined statement (f) can be true if one is true and the other is false. Thus, a true statement and a false statement separated by the connective OR make a true statement (if A and/or B is true). If *neither* the tape frame pulse nor the edit pulse is present (A = B= false = 0), then the statement is false.

The Exclusive OR

Now make two statements:

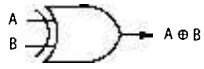
A. The main transmitter will feed the antenna.

OR ←——connective

B. The standby transmitter will feed the antenna.

This combined statement implies that either A OR B, but not both, can be true. Thus the Exclusive OR function is required.

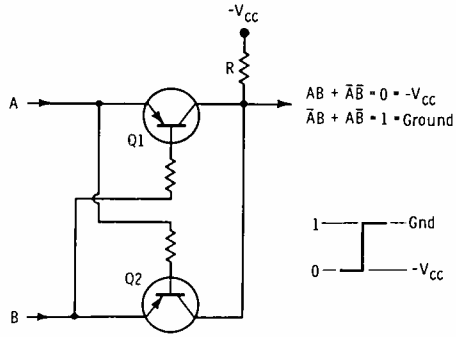
The symbol for the Exclusive OR is: A ⊕ B. The Exclusive OR schematic symbol is shown in Fig. 3-7A. The Exclusive OR truth table is presented in Fig. 3-7B. Note that to produce an output (1), the inputs must be opposite. Like inputs (either 0 or 1) result in no



(A) *Schematic symbol.*

A	B	f = A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

(B) *Truth table.*



(C) *Discrete circuit.*

Fig. 3-7. Exclusive OR function.

output (0). Thus, $\bar{A}B$ OR $A\bar{B}$ (read "A NOT AND B OR A AND B NOT") is required for an output. In symbols:

$$A \oplus B = \bar{A}B + A\bar{B}$$

Fig. 3-7C shows a discrete circuit that illustrates the Exclusive OR function. Although IC chips are normally used, this discrete circuit allows the basic action to be illustrated. If both A and B are of like polarity (0's or 1's), both transistors have zero-biased base-emitter junctions, and neither can conduct. Thus, the output is the full negative value of $-V_{CC}$. Now assume A is 0 (ground) and B is +5 volts ($\bar{A}B$). Thus, the base of Q2 is grounded and the emitter is at +5 volts, causing Q2 to conduct. This sends the output essentially to ground, creating the "high" (ground) condition and producing a 1 output. If A is +5 volts and B is grounded (the condition for $A\bar{B}$), then Q1 conducts, also producing a 1 output. Thus, the condition for $\bar{A}B + A\bar{B} = 1$ is satisfied.

3-5. VARIATIONS ON THE CONNECTIVE AND

Up to this point, every *statement* (or *proposition*) has been positive in content. When you say, "the oscillator output level is normal, and the frequency is within tolerance," you are making a positive statement that is either true or not true (false).

Every statement has an opposite:

A = The oscillator output level is normal.
$\bar{A}$ = The oscillator output level is NOT normal.
B = The frequency is within tolerance.
$\bar{B}$ = The frequency is NOT within tolerance.

The bar over the A ($\bar{A}$) indicates the negative, or inverted form, of A. Similarly, the bar over the B ($\bar{B}$) indicates the negative, or inverted form, of B.

If A = 1, $\bar{A}$ = 0. (If A is high, $\bar{A}$ is low.)
If A = 0, $\bar{A}$ = 1. (If A is low, $\bar{A}$ is high.)
If B = 1, $\bar{B}$ = 0. (If B is high, $\bar{B}$ is low.)
If B = 0, $\bar{B}$ = 1. (If B is low, $\bar{B}$ is high.)

Sometimes a "straight inverting" stage (NOT gate) is used for logic inversion. The symbol is shown in Fig. 3-8A. The circle at the

(A) *High-level activated.* (B) *Low-level activated.*

Fig. 3-8. Schematic symbols for NOT circuits.

output means a 1 input (high-level) results in a 0 (low-level) output. This is termed a high-level activated stage.

If the circle appears at the input (Fig. 3-8B), a 0 (low) input results in a 1 (high) output. This is a low-level activated stage. Either type of NOT gate is inverting. The circle at the input or output is important only to indicate clearly the significant function.

Inverters are generally contained in ICs. For example, the Type 7404 is a hex inverter, containing six independent inverter stages in one IC.

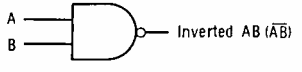
A NOT AND circuit is termed a NAND gate. The schematic symbol is shown in Fig. 3-9A. The notation $\overline{AB}$ is read "NOT A AND B." The small circle at the output indicates polarity reversal. This means that if A and B are both true (1), the output ($\overline{AB}$) is false (0). It is the inverted AND function.

Fig. 3-9B shows how an ordinary NAND gate can be connected as a straight inverter. Since both inputs are tied together, a 1 at the input will result in a 0 (inverted) output.

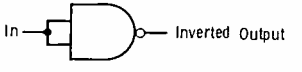
IMPORTANT NOTE: If both inputs of a NAND gate are "floating" (neither ground nor a positive voltage), internal connections result in a positive voltage (1.2 to 1.8 volts) at both inputs, resulting in a zero (low) voltage at the output. Therefore, the inputs must swing between ground (or a negative voltage) and a positive voltage. An open on either input will cause the NAND gate to be "stuck" in the low mode, providing the other input is high. This feature is common *to all gates;* an open input will generally be interpreted as a *high level.*

Fig. 3-9C shows the truth table for the AND function and the NAND function. Note that if one input of a NAND gate is 0, the output will always be 1. The output is the reverse of AB.

The *inhibitor* gate produces an output only when the inputs represent $A\overline{B}$ (A AND B NOT) or $\overline{A}B$ (A NOT AND B). Fig. 3-10A shows the schematic symbol for $f = A\overline{B}$. The $\overline{B}$ may be obtained from a straight inverter stage preceding the B input.



(A) *Schematic symbol.*

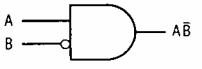


(B) *Connected as inverter.*

A	B	AB	$\overline{AB}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

(C) *Truth table.*

Fig. 3-9. The NAND gate.



(A) *Schematic symbol.*

A	B	$A\overline{B}$	$\overline{A}B$
0	0	0	0
0	1	0	1
1	0	1	0
1	1	0	0

(B) *Truth table.*

Fig. 3-10. Inhibitor gate.

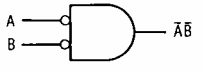
The small circle at the B input represents inversion. A 1 input at B becomes a 0 input to the AND gate. A 0 input at B becomes a 1 input to the AND gate. Thus, $f = A\overline{B}$ (read as "A AND B NOT"). For a 1 to occur at the output, A must be 1 and B must be 0. Obviously, if the circle were on the A input, $f = \overline{A}B$ (read as "A NOT AND B"), and the conditions would be reversed.

Fig. 3-10B shows the truth table for $A\overline{B}$, $\overline{A}B$. Note that for $f = A\overline{B}$, the output is inhibited (0) for all conditions except when $A = 1$ and $B = 0$. For $f = \overline{A}B$, the output is inhibited except when $A = 0$ and $B = 1$. In either case, the inputs must be of opposite character for an output to occur.

It was stated in Section 3-3 that $\overline{A}$ can be read either "NOT A" or "A NOT" and that $\overline{B}$ can be read either "NOT B" or "B NOT." The "A NOT" and "B NOT" reading is preferred, as the following example shows.

Assume you have the condition of $\overline{A}B$. If you read this as "NOT A AND B," you are stating the condition of $\overline{AB}$. But note the decided difference between the truth tables in Figs. 3-9C and 3-10B. For this reason, you should use the phrase "A NOT AND B" for $f = \overline{A}B$.

When small circles (indicating inversion) are at both inputs of an AND gate, as in Fig. 3-11A, $f = \overline{A}\overline{B}$ is read as f = A NOT AND B NOT. Fig. 3-11B shows the truth table for $\overline{A}\overline{B}$. The output function is a 1

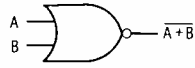


(A) *Schematic symbol.*

A	B	$\overline{A}\overline{B}$
0	0	1
0	1	0
1	0	0
1	1	0

(B) *Truth table.*

Fig. 3-11. Characteristics of A NOT AND B NOT gate.



(A) *Schematic symbol for NOR gate.*

A	B	A+B	$\overline{A+B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

(B) *Truth table for OR and NOR.*

Fig. 3-12. The NOR gate.

when (and only when) both inputs are 0. Note carefully that $\overline{A}\overline{B}$ (A NOT AND B NOT) is different from $\overline{AB}$ (NOT A AND B). Review Fig. 3-9C.

3-6. VARIATIONS ON THE CONNECTIVE OR

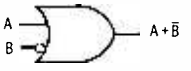
Inversion is used in OR gates as well as in AND gates. A NOT OR gate is called a NOR gate. The schematic symbol is shown in Fig. 3-12A. The notation $\overline{A + B}$ is read as "NOT A OR B." The small circle at the output indicates polarity inversion. This is a reversed-polarity OR gate. Only when both inputs are low (0) is the output high (1). Fig. 3-12B is a truth table comparing $A + B$ and $\overline{A + B}$. (OR and NOR gates).

Three-state (sometimes called *trilevel*) logic devices have an extra input termed an *enable-disable* gating input. When the gate is *enabled*, the output is either logic 0 or logic 1, but not both. When it is *disabled*, the output is disconnected from the rest of the circuit. Fig. 3-13A illustrates the extra input that *enables* the gate with a logic 0. Fig. 3-13B shows the extra input that *enables* the circuit with a logic 1.

IMPORTANT NOTE: Three-state, or three-level, logic devices are not limited to NOR gates. In practice, AND, NAND, OR, and NOR gates as well as straight inverter and noninverter stages are available with enable/disable inputs. (The name Tri-State is a registered trademark of National Semiconductor Corp.)

(A) *Logic 0 enables.* (B) *Logic 1 enables.*

Fig. 3-13. Symbols for three-state NOR gates.



(*A*) *Schematic symbol.*

A	B	$\bar{A}$	$\bar{B}$	$A+\bar{B}$	$\bar{A}+B$
0	0	1	1	1	1
0	1	1	0	0	1
1	0	0	1	1	0
1	1	0	0	1	1

(*B*) *Truth table.*

Fig. 3-14. Gate for A OR B NOT function.

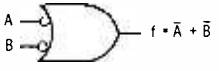
With the circuit of Fig. 3-14A, an ORed output can be obtained under all conditions except when A = 0 and B = 1. The notation $A + \overline{B}$ is read as "A OR B NOT." The small circle on the B input indicates inversion. A 1 at the B input becomes a 0 at the OR gate input. A 0 at the B input becomes a 1 at the OR input. If the circle were at the A input, then the function would be $\overline{A} + B$ (read as "A NOT OR B"). Fig. 3-14B is the truth table for $A + \overline{B}$, $\overline{A} + B$.

For clarity, the inverted values of A and B are included in Fig. 3-14B:

$$A = 1, \overline{A} = 0$$
$$A = 0, \overline{A} = 1$$
$$B = 1, \overline{B} = 0$$
$$B = 0, \overline{B} = 1$$

When there are small circles (indicating inversion) on both inputs of an OR gate (Fig. 3-15A), the function is $f = \overline{A} + \overline{B}$ ("A NOT OR B NOT"). Only when both inputs are high (1) is the output low (0). Fig. 3-15B is the truth table for $\overline{A} + \overline{B}$. Compare this truth table with Fig. 3-12B. Note that $\overline{A} + \overline{B}$ is *not* the same as $\overline{A + B}$.

When an OR gate is specified, the Inclusive OR is implied. The schematic symbol for the inverted Exclusive OR (Exclusive NOR) is

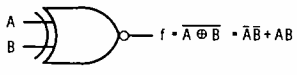


(*A*) *Schematic symbol.*

A	B	$\bar{A}+\bar{B}$
0	0	1
0	1	1
1	0	1
1	1	0

(*B*) *Truth table.*

Fig. 3-15. Gate for A NOT OR B NOT.



(*A*) *Schematic symbol.*

A	B	$\bar{A}\bar{B}$ + AB
0	0	1
0	1	0
1	0	0
1	1	1

(*B*) *Truth table.*

(*C*) *Discrete circuit.*

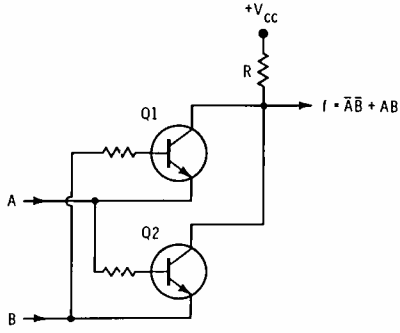


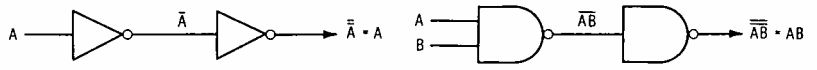
Fig. 3-16. Exclusive NOR gate.

shown in Fig. 3-16A. The Exclusive NOR truth table is shown in Fig. 3-16B. Note that: $\overline{A \oplus B} = \bar{A}\bar{B} + AB$ is "A NOT AND B NOT OR A AND B."

A discrete circuit that produces an output of $\overline{A \oplus B}$ is shown in Fig. 3-16C. When A and B are alike (either 0 or 1) Q1 and Q2 have zero-biased base-emitter junctions and form open switches. With no current through R, $f = V_{CC}$ = high = 1. When A and B differ (0, 1), the opposite logic levels cause either Q1 or Q2 to conduct. The resultant current in R sends the output to essentially ground potential (low level, or 0). Note that this is just the inverse of the Exclusive OR gate, which requires *unlike* input polarities to produce an output (Fig. 3-7C). Note also that Figs. 3-7C and 3-16C are simply inverse circuits; the type of transistors (npn or pnp) and voltage polarity are reversed.

3-7. EQUIVALENCES (DUALITY) OF LOGIC GATES

When a given character is passed through one inverting stage, the polarity of the original character is reversed. When a given character is passed through two consecutive inverting stages, the output has the original character polarity. See Fig. 3-17A, where $\bar{A}$ = in-



(*A*) *Double inversion of A.* (*B*) *Double inversion of AB.*

Fig. 3-17. Examples of double inversion.

verted A and $\bar{\bar{A}}$ = double inverted A = A. This is the same as passing any signal through two common-cathode (tube) stages or common-emitter (transistor) stages; the original polarity results. Note that the same holds true for $\overline{\overline{AB}} = AB$ (Fig. 3-17B).

Fig. 3-18 is a comparative truth table that lists many of the logic circuits covered thus far. Pick out any duplication in this truth table, and write the resulting relationship:

Note that $\overline{AB}$ and $\bar{A} + \bar{B}$ have the same function. Therefore, $\overline{AB}$ is equivalent to $\bar{A} + \bar{B}$. (Read as NOT A AND B is equivalent to A NOT OR B NOT.") Thus, a duality exists for $\overline{AB}$ and $\bar{A} + \bar{B}$.

Also note that $\bar{A}\bar{B}$ has the same function as $\overline{A + B}$. Therefore, $\bar{A}\bar{B}$ is equivalent to $\overline{A + B}$ (read as "A NOT AND B NOT is equivalent to NOT A OR B"). Thus, a duality exists for $\bar{A}\bar{B}$ and $\overline{A + B}$.

Fig. 3-19 reviews the truth tables and symbols for logic gates and shows the dual nature of such gates. Study this figure, making sure you understand the duality of the symbols shown.

3-8. LOGIC NETWORKS

Logic networks (combined AND, NAND, OR, and NOR gates) are used to satisfy Boolean relationships. In the design of logic systems, the functions are first reduced to Boolean form. Fig. 3-20 illustrates an example of a logic network designed to perform the Exclusive OR function. Note that $f = \bar{A}B + A\bar{B} = A \oplus B$ = Exclusive OR.

Although multiple components can be and sometimes are used to perform this function, complete Exclusive OR gates fabricated on an integrated-circuit chip are available. Examples are Motorola quad Exclusive OR Type MC 14507CL or MC 771P and Signetics Type N 7486.

A	B	AB	$\overline{AB}$	$\bar{A}\bar{B}$	A+B	$\overline{A+B}$	$\bar{A}+\bar{B}$
0	0	0	1	1	0	1	1
0	1	0	1	0	1	0	1
1	0	0	1	0	1	0	1
1	1	1	0	0	1	0	0

Fig. 3-18. Comparison of logic gates showing duality (see text).

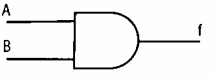
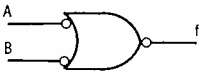
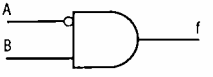
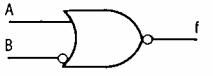
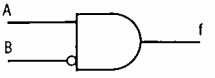
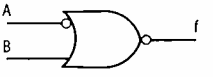
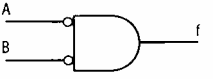
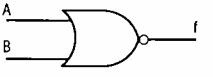
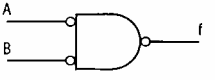
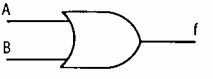
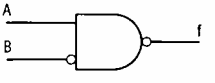
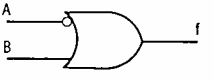
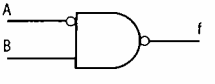
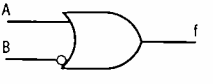
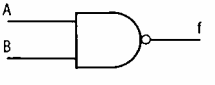
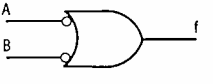
Row	AND (NAND)	Truth Table	OR (NOR)
1		A B f 0 0 0 0 1 0 1 0 0 1 1 1	
2		A B f 0 0 0 0 1 1 1 0 0 1 1 0	
3		A B f 0 0 0 0 1 0 1 0 1 1 1 0	
4		A B f 0 0 1 0 1 0 1 0 0 1 1 0	
5		A B f 0 0 0 0 1 1 1 0 1 1 1 1	
6		A B f 0 0 1 0 1 1 1 0 0 1 1 1	
7		A B f 0 0 1 0 1 0 1 0 1 1 1 1	
8		A B f 0 0 1 0 1 1 1 0 1 1 1 0	

Fig. 3-19. Common logic symbols (gates).

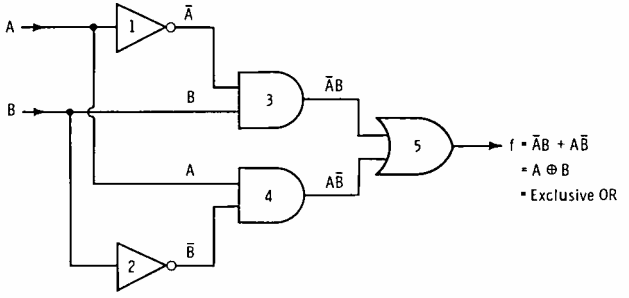


Fig. 3-20. Example of logic network.

With the most significant digit (MSD) on the left, the basic binary-to-Boolean table for up to four variables is shown in Fig. 3-21. Letters A, B, C, and D each must have either a 0 or 1 value. Thus, where $n = 1$ and $\bar{n} = 0$ (n = any letter):

$$\text{Binary } 01 = \bar{A}B \text{ Boolean}$$
$$10 = A\bar{B} \text{ Boolean}$$
$$101 = A\bar{B}C \text{ Boolean}$$
$$1010 = A\bar{B}C\bar{D} \text{ Boolean}$$
$$\text{etc.}$$

The last letter (B, C, or D) is the least significant digit (LSD). The first letter (A) is the most significant digit (MSD).

NOTE: Some digital functions specify A as the *least significant bit*, with the last letter used specified as the *most significant bit*. The arrangement used is immaterial provided the nomenclature is understood for any specific application.

It is important to realize that AND, OR, NAND, and NOR gates are not restricted to two inputs. Integrated-circuit gates with five or more inputs are available.

Table 3-1 lists all the binary and Boolean relationships, with rule numbers for future reference.

Binary					Base 10 Max Count	Modulus (Total No. of Levels)
Weight *	8	4	2	1		
2 - Variable			A	B	2 + 1 = 3	$2^2 = 4$
3 - Variable		A	B	C	4 + 2 + 1 = 7	$2^3 = 8$
4 - Variable	A	B	C	D	8 + 4 + 2 + 1 = 15	$2^4 = 16$

*See NOTE in text.

Fig. 3-21. Binary-Boolean table.

Table 3-1. Binary and Boolean Relationships

Rule No.	Relation	Laws	
1	$0 + 0 = 0$	Binary	
2	$(0)(0) = 0$		
3	$1 + 1 = 1$		
4	$(1)(1) = 1$		
5	$(0)(1) = 0$		
6	$0 + 1 = 1$		
7	$\overline{0} = 1$		
8	$\overline{1} = 0$		
9	$A + A = A$	Boolean-Binary	
10	$(A)(A) = A$		
11	$(\overline{A})(A) = 0$		
12	$\overline{A} + A = 1$		
13	$0 + A = A$		
14	$(0)(A) = 0$		
15	$1 + A = 1$		
16	$(1)(A) = A$		
17	$A + B = B + A$	Commutative	Three basic algebraic laws applicable to Boolean expressions
18	$(A)(B) = (B)(A)$		
19	$(A + B) + C = A + (B + C)$	Associative	
20	$(AB)C = A(BC)$		
21	$A(B + C) = AB + AC$	Distributive	
22	$A + AB = A$	Simplification Rules (DeMorgan's Theorem)	
23	$A + \overline{A}B = A + B$		
24	$A(A + B) = A$		
25	$\overline{AB} = \overline{A} + \overline{B}$		
26	$\overline{A + B} = \overline{A}\,\overline{B}$		

NOTE: Multiple letters within parentheses must be considered as a group. Thus, for A(A + B) the group (A + B) is either a 0 or a 1. The relation A(A + B) is read "A and the quantity A OR B." This network would be an AND/OR combination.

Fig. 3-22 proves Rule 24, that $A(A + B) = A$, through the use of switch contacts. Note that the output function depends entirely on the 0 or 1 position of A, regardless of the B mode.

To prove that $A(A + B) = A$ by algebraic means:

$A(A + B) = AA + AB$	Rule 21
$AA + AB = A + AB$	Rule 10
$A + AB = (1)(A) + AB$	Rule 16
$(1)(A) + AB = A(1 + B)$	Rule 21
$A(1 + B) = (A)(1)$	Rule 15
$(A)(1) = A$	Rule 16
Therefore $A(A + B) = A$	

Fig. 3-23 proves that $A(A + B) = A$ by the truth-table method. The steps in the truth-table form of simplification or demonstration are:

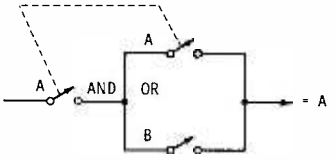


Fig. 3-22. Use of switch contacts to prove relationship A(A + B) = A.

1. List the variables and all combinations called for in the expression to be simplified (Row 1, columns 1 through 4 in this example).
2. Put down all possible binary values of the elementary letters (A and B in this example, rows 2 through 5, columns 1 and 2).
3. Determine the values of the elementary combinations called for in the expression to be simplified (A + B in this example, rows 2 through 5).
4. List the values of the combinations of combinations (rows 2 through 5, column 4 in this example).

Note that the columns for A and A(A + B) are identical. Since the A + B part of A(A + B) is superfluous, the simplest term is A. This means that A(A + B) = A.

For another example, simplify $A\overline{B}\overline{C} + ABC + \overline{A}BC$ algebraically.

$$A\overline{B}\overline{C} + ABC + \overline{A}BC = A\overline{B}\overline{C} + BC(A + \overline{A}) \quad \text{Rule 21}$$

$$= A\overline{B}\overline{C} + BC \quad \text{Rule 12}$$

It is imperative that the simplest relationship possible be used in logic design. Sometimes it is not readily evident when final simplification has been reached algebraically. *Karnaugh maps* or *Veitch diagrams* are often used. You are encouraged to study Karnaugh maps (covered in many book-length treatments of logic theory), particularly if you desire to enter the field of digital design.

The chart form of simplification can be used as in Fig. 3-24. The example shown is for three variables, but as many variables as necessary can be handled in chart form. Note that all possible combinations of A, B, and C must be accounted for. The procedure for simplifying $A\overline{B}\overline{C} + ABC + \overline{A}BC$ follows:

Column → / Row ↓	1	2	3	4
1	A	B	A+B	A(A+B)
2	0	0	0	0
3	0	1	1	0
4	1	0	1	1
5	1	1	1	1
		Identical		

Fig. 3-23. Proof of relationship A(A + B) = A by truth-table method.

Column	1	2	3	4	5	6	7
Row	A	B	C	AB	AC	BC	ABC
1	~~$\bar{A}$~~	~~$\bar{B}$~~	~~$\bar{C}$~~	~~$\bar{A}\bar{B}$~~	~~$\bar{A}\bar{C}$~~	~~$\bar{B}\bar{C}$~~	~~$\bar{A}\bar{B}\bar{C}$~~
2	~~$\bar{A}$~~	~~$\bar{B}$~~	~~C~~	~~$\bar{A}\bar{B}$~~	~~$\bar{A}$C~~	~~$\bar{B}$C~~	~~$\bar{A}\bar{B}$C~~
3	~~$\bar{A}$~~	~~B~~	~~$\bar{C}$~~	~~$\bar{A}$B~~	~~$\bar{A}\bar{C}$~~	~~B$\bar{C}$~~	~~$\bar{A}$B$\bar{C}$~~
4	~~$\bar{A}$~~	~~B~~	~~C~~	~~$\bar{A}$B~~	~~$\bar{A}$C~~	(BC)	~~$\bar{A}$BC~~
5	~~A~~	~~$\bar{B}$~~	~~$\bar{C}$~~	~~A$\bar{B}$~~	~~A$\bar{C}$~~	~~$\bar{B}\bar{C}$~~	($A\bar{B}\bar{C}$)
6	~~A~~	~~$\bar{B}$~~	~~C~~	~~A$\bar{B}$~~	~~AC~~	~~$\bar{B}$C~~	~~A$\bar{B}$C~~
7	~~A~~	~~B~~	~~$\bar{C}$~~	~~AB~~	~~A$\bar{C}$~~	~~B$\bar{C}$~~	~~AB$\bar{C}$~~
8	~~A~~	~~B~~	~~C~~	~~AB~~	~~AC~~	(BC)	~~ABC~~

Simplify: $A\bar{B}\bar{C} + ABC + \bar{A}BC$
Solution: $A\bar{B}\bar{C} + BC$

Fig. 3-24. Example of chart form of simplification.

1. Look at column 7. Draw a line through each complete row whose term in column 7 is *not* contained in the expression to be simplified (rows 1, 2, 3, 6, and 7 in this example).
2. Starting with column 1, cross out all terms identical to those that were lined out in the same column in Step 1. For example, in column 1, $\bar{A}$ is crossed out in row 4, and A is crossed out in rows 5 and 8. In column 2, $\bar{B}$ is crossed out in row 5, and B is crossed out in rows 4 and 8. As the same procedure is followed for the remaining columns in this example, all terms in columns 1, 2, 3, 4, and 5 are crossed out.
3. In column 6, BC (rows 4 and 8) is not eliminated. Circle these for identification.
4. Starting at the left, go to the right and rule out all terms containing BC in all rows that contain a circled BC (in this example, rows 4 and 8, column 7).
5. All terms are now ruled out except BC in column 6 and $A\bar{B}\bar{C}$ in column 7. Only $A\bar{B}\bar{C}$ and BC remain; therefore, $A\bar{B}\bar{C} + ABC + \bar{A}BC = A\bar{B}\bar{C} + BC$.

SUMMARY: Boolean relationships can be simplified by several means: algebraic relationships, switch contacts, truth tables, Karnaugh maps, or charts. The Karnaugh map is not covered in this book, since a really practical presentation is quite lengthy, and it is useful primarily to design engineers.

Rules 25 and 26 in Table 3-1 (DeMorgan's theorem) illustrate the *duality* of Boolean (hence logic) relationships. In words:

1. The truth value of statement A OR B is identical to the truth value of the *negation* of the statement A NOT AND B NOT. In symbols $A + B = \overline{\bar{A}\bar{B}}$.
2. The truth value of statement A AND B is identical to the *negation* of the statement A NOT OR B NOT. In symbols: $AB = \overline{\bar{A} + \bar{B}}$.

Column	1	2	3	4	5	6	7	8	9	10	11	12
Row 1	A	B	$\overline{A}$	$\overline{B}$	AB	$\overline{AB}$	$\overline{A}\overline{B}$	$\overline{\overline{A}\overline{B}}$	A+B	$\overline{A+B}$	$\overline{A}+\overline{B}$	$\overline{\overline{A}+\overline{B}}$
2	0	0	1	1	0	1	1	0	0	1	1	0
3	0	1	1	0	0	1	0	1	1	0	1	0
4	1	0	0	1	0	1	0	1	1	0	1	0
5	1	1	0	0	1	0	0	1	1	0	0	1

Fig. 3-25. Proof of DeMorgan's theorem by truth-table method.

This can be proved by the truth table of Fig. 3-25. We will repeat here the rules for construction of such a truth table as applied specifically to Fig. 3-25:

1. List the variables and all possible combinations (row 1, columns 1 through 12 in this example).
2. Determine the values of elementary letters (rows 2 through 5, columns 1 through 4 in this example).
3. From Step 2, determine the values of combinations (rows 2 through 5, columns 5 through 12 in this example).

Note that $A + B = \overline{\overline{A}\overline{B}}$

$$AB = \overline{\overline{A} + \overline{B}}$$

$$\overline{AB} = \overline{A} + \overline{B} \quad \text{(Rule 25)}$$

$$\overline{A}\overline{B} = \overline{A + B} \quad \text{(Rule 26)}$$

Restate DeMorgan's theorem as follows: If every + is replaced by a $(\cdot)$ and every $(\cdot)$ is replaced by a +, and each variable is replaced by its inverted value, then the resultant function (f) equals the original function inverted.

NOTE: In our treatment, the symbol $(\cdot)$ is implied, where $AB = A \cdot B = A$ AND B.

Given: $f = AB$. Derive DeMorgan's theorem, Rule 25.
Step 1. Replace implied $(\cdot)$ with +: $A + B$.
Step 2. Invert each variable: $\overline{A} + \overline{B}$.
Step 3. Then $f = \overline{A} + \overline{B} = \overline{AB}$. (AB inverted $= \overline{AB}$.) This is derived Rule 25, illustrated by Fig. 3-19, row 8.
Given $f = A + B$. Derive DeMorgan's theorem, Rule 26.
Step 1. Replace + with implied $(\cdot)$: AB.
Step 2. Invert each variable: $\overline{A}\overline{B}$.
Step 3. Then $f = \overline{A}\overline{B} = \overline{A + B}$. ($A + B$ inverted is $\overline{A + B}$.) This is derived Rule 26, illustrated by Fig. 3-19, row 4.

3-9. NEGATIVE LOGIC

The logic discussed so far assumes a 1 to be represented by a more positive level than a 0, as in Fig. 3-26A. It is often necessary to perform an operation when a function is absent (0). Negative logic (Fig. 3-26B) is used to provide a signal when a function is absent. In negative logic, the most *negative* voltage level represents logical 1.

Connect two diodes and a resistor as in Fig. 3-27A. Note that for negative logic, both diode inputs (A AND B) must be at 0 volts (ground or logical 1) for a logical 1 (0 volts) to appear at f. In symbols, f = AB (AND gate). For positive logic, a logical 1 (+5 volts) at either input results in f = 1. In symbols, f = A + B (OR gate). For negative logic:

$$0 \text{ volts} = \text{Boolean } 1$$
$$+5 \text{ volts} = \text{Boolean } 0$$

For positive logic:

$$+5 \text{ volts} = \text{Boolean } 1$$
$$0 \text{ volts} = \text{Boolean } 0$$

RULE: A positive-logic OR gate is a negative-logic AND gate.

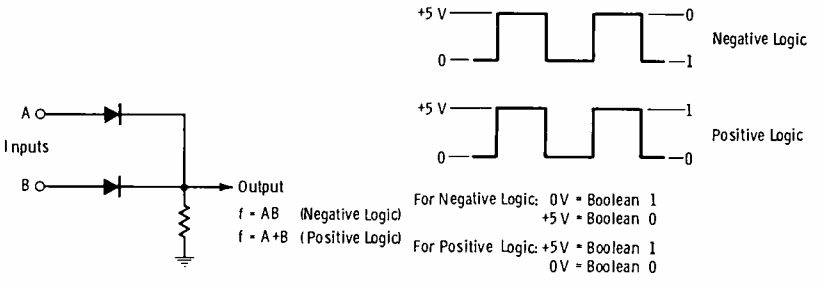
Connect two diodes and a resistor as in Fig. 3-28A. For negative logic, either diode with a 0 (ground) input (logical 1) results in essentially 0 volts (logical 1) at f. In symbols, f = A + B (OR gate). For positive logic, *both* inputs (A AND B) must be at +5 volts (logical 1) for +5 volts (logical 1) to appear at f. (The opposite diode input is assumed to remain at ground when just one input goes to +5 volts.) In symbols, f = AB (AND gate).

RULE: A positive-logic AND gate is a negative-logic OR gate. This is a dual nature of gates that depends on the input logic chosen. For fixed logic, once this logic level is chosen, the duality vanishes.

Negative logic and positive logic are often performed in the same circuitry. For example, if positive-logic OR and NOR functions are being performed using OR and NOR ICs, negative-logic AND or NAND functions can be performed with the same ICs. Many gates come in quads (four complete gates in one IC). Thus, if two positive-

(A) *Positive logic.* (B) *Negative logic.*

Fig. 3-26. Comparison of positive and negative logic.



(A) *Diode circuit.* (B) *Logic definitions.*

Fig. 3-27. Diode positive (OR) and negative (AND) logic.

logic OR gates, for example, are used, two negative-logic AND gates are still available in the same IC if required. This greatly reduces the number of components needed for given applications.

Given the positive-logic truth table of Fig. 3-29A, write the corresponding truth table for negative logic as in Fig. 3-29B. Note that you simply change all zeros to ones and all ones to zeros. In Fig. 3-29A, therefore, $f = AB$ (AND gate for positive logic). In Fig. 3-29B, the corresponding negative logic shows an OR gate ($f = A + B$).

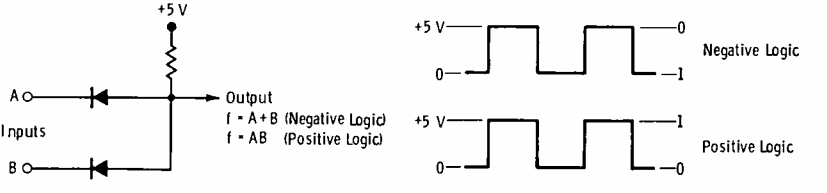
Given the positive-logic truth table for $\overline{A + B}$ as in Fig. 3-30, write the corresponding negative-logic truth table. Positive logic $\overline{A + B} = \overline{A}\,\overline{B}$ (Rule 26, Table 3-1, and Fig. 3-18). The corresponding negative logic is $\overline{A}\,\overline{B} = \overline{A} + \overline{B}$ (Rule 25, Table 3-1, and Fig. 3-18).

If you follow the same line of reasoning: Positive logic $\overline{A}\,\overline{B} = \overline{A} + \overline{B}$ (Rule 25, Table 3-1, and Fig. 3-18), and the corresponding negative logic is $\overline{A} + \overline{B} = \overline{A}\,\overline{B}$ (Rule 26, Table 3-1, and Fig. 3-18).

NOTE: All logic is assumed to be positive unless specifically identified as negative logic.

3-10. LOGIC FAMILIES

Discrete transistor circuitry and logic ICs are classified into various "families," such as RTL, DTL, etc. The basic characteristics of these families may be outlined as follows.



(A) *Diode circuit.* (B) *Logic definitions.*

Fig. 3-28. Diode positive (AND) and negative (OR) logic.

A	B	f = AB
0	0	0
0	1	0
1	0	0
1	1	1

(*A*) *Positive logic* (*AND*).

A	B	f = A+B
1	1	1
1	0	1
0	1	1
0	0	0

(*B*) *Negative logic* (*OR*).

Fig. 3-29. Truth tables for positive-logic function and corresponding negative-logic function.

RTL (*Resistor-Transistor Logic—Fig. 3-31A*): Logic performed by resistors provides straightforward circuit design. This type is relatively slow in operation.

RCTL (*Resistor-Capacitor-Transistor Logic—Fig 3-31B*): Capacitors are added around the resistors to increase the switching speed. This type is faster than RTL.

DCTL (*Direct-Coupled Transistor Logic—Fig. 3-31C*): Logic is performed by transistors. This type provides relatively fast switching speeds, high power efficiency, and better miniaturization.

TTL (*Transistor-Transistor Logic—Fig. 3-31D*): Logic is performed by transistors at high speed. In Fig. 3-31D, Q1 and Q2 are connected as common-base amplifiers. The collectors of Q1 and Q2 are paralleled and connected to the base of Q3, the output transistor, which operates as an inverting amplifier. If the emitters of Q1 and Q2 are both at high level, the transistors are cut off and the collectors are essentially at the high base voltage. The resulting current through D1 and R1 sends the base of Q3 positive, saturating that transistor and pulling the output almost to ground potential. If signal A or B is zero volts, either Q1 or Q2 will be on and saturated. This brings the paralleled collectors to approximately 0.6

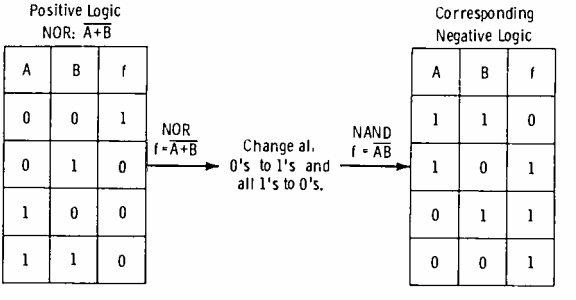


Fig. 3-30. Truth-table form of demonstrating that positive-logic $\overline{A + B}$ is equivalent to negative-logic $\overline{AB}$.

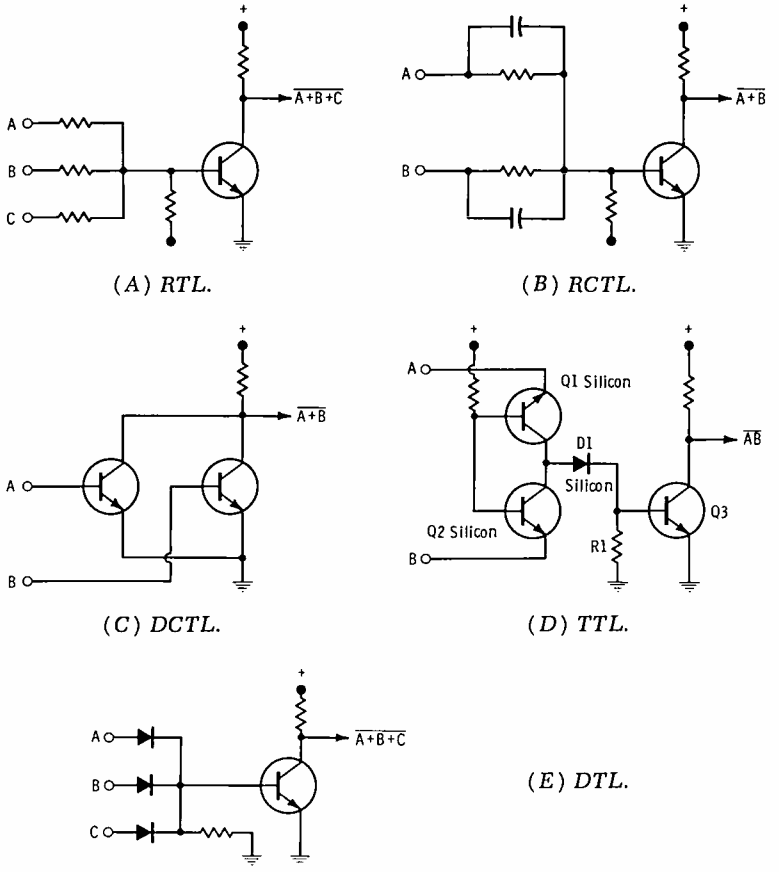


Fig. 3-31. Basic logic families.

volt (assuming silicon transistors), and since diode D1 is also silicon, the base of Q3 is essentially at ground potential. Thus, Q3 is cut off and a "high" output occurs.

DL (Diode Logic—Figs. 3-27 and 3-28): Logic is performed by diodes. High speeds are possible with this type. The signal is noninverted. Diode logic is relatively inexpensive, but amplifiers may be required to maintain correct logic levels through several cascaded gates.

DTL (Diode-Transistor Logic—Fig. 3-31E): Logic is performed by diodes. The output is inverted. Small voltage excursions minimize the effects of stray capacitance and thus increase switching speed.

MOS (Metal-Oxide-Semiconductor) Logic: The MOS configuration is generally applied to ICs. These are extremely high-impedance,

low-current devices, with excellent noise immunity. They may operate at voltages up to around 18 volts, as compared with the general practice of using a logic level of 5 volts in other logic circuitry. In general, MOS ICs employ a field-effect transistor (FET) structure internally.

ECL (Emitter-Coupled Logic): Emitter-coupled logic is just what the name implies. The low internal impedance of an emitter follower provides excellent immunity from capacitive effects; hence, high-speed switching is possible.

3-11. INTERFACING LOGIC FAMILIES

All logic devices have many specifications in addition to the AND, OR, NAND, or NOR gate function. These can be obtained only from the specification sheet for the individual gate. For example, an "open collector" may be specified (see Fig. 3-35B). This means that the gate can be used for applications in which many collectors or other components are to be tied together. In addition, all logic families have certain general characteristics that may not be compatible with other gates—the magnitude of V_{CC} (or V_{DD}), pulse levels (affecting noise immunity), speed requirements, etc.

Fig. 3-32 is an example of logic-level characteristics at the input of a gate. In this example, 0 to 1 volt is the logic 0 input region. The 1-volt level is the maximum input at low level for which the output logic level does not change state. The region from 1 volt to 3.5 volts is an indeterminate region; this specific gate is not guaranteed to change states in that region. The level at 3.5 volts is the guaranteed level for a logical 1 output, and it is the minimum input at high level for which the output does not change state after being in the logical 1 state.

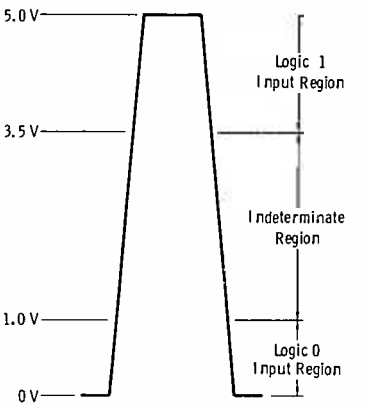


Fig. 3-32. Example of logic-level characteristics at device input.

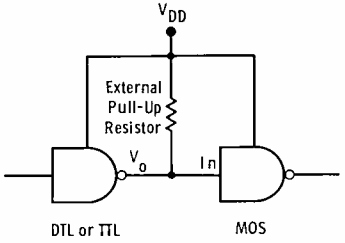


Fig. 3-33. Typical interface for DTL or TTL to MOS.

The above example is a relatively *high-threshold* logic. Both DTL and TTL circuitry have relatively lower logic levels. When logic devices of different families are interfaced, these characteristics must be borne in mind. For example, a typical minimum logic-1 TTL output is 3.6 volts. If this device is to be used to drive an MOS circuit (minimum logic-1 input voltage of 3.5 volts), the noise margin is only $3.6 - 3.5 = 0.1$ volt, which is a borderline situation.

The typical solution to this type of interfacing problem is illustrated in Fig. 3-33. An external *pull-up resistor* is used so that V_o is equal to V_{DD}. This resistor affects the logic-0 interface condition so that V_o maximum is increased. Determination of the optimum value requires consideration of fanout (discussed in next section), maximum DTL or TTL device current, V_o maximum, TTL device leakage in the high state, power consumption, and propagation delay. There is no substitute for the specification sheet and the manufacturer's application notes for interfacing information.

The RCA COS/MOS buffer Type CD4009A is designed as a level shifter (Fig. 3-34). This permits operation of MOS logic at supply-voltage values up to +15 volts but also makes possible direct interface with TTL or DTL devices at +5 volts. The system advantage of this scheme is the increased logic speed of COS/MOS circuits when operated at voltages higher than +5 volts.

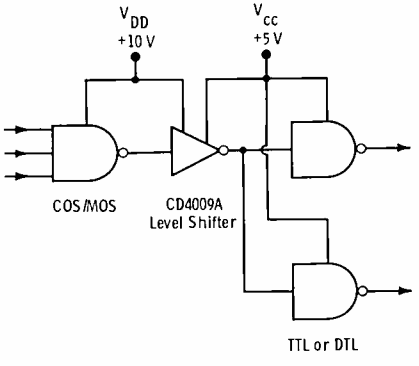
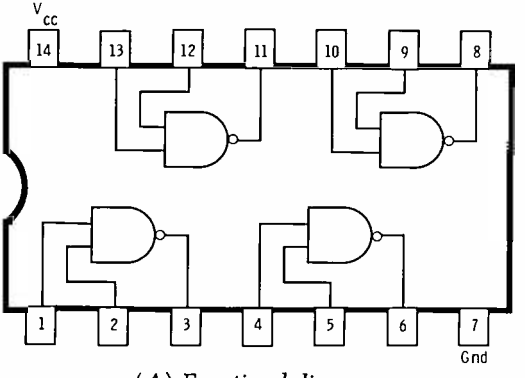


Fig. 3-34. Logic interface for logic-level conversion.

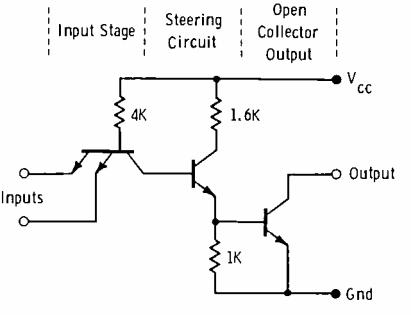
The *open collector* configuration allows easy interfacing without special arrangements. The functional makeup of the Type 7426, a quad two-input NAND gate, is shown in Fig. 3-35A. The schematic diagram of an individual gate is shown in Fig. 3-35B. The collector supply voltage is a nominal +5 volts. The open collector with high-voltage capability (15 volts) permits interfacing with MOS, lamps, or relays. (*Caution:* All "open collector" ICs do not have the high-voltage feature.) When an IC, such as the Type 7421, does not have an open collector (Fig. 3-36), a *totem-pole* arrangement is normally used for the output as illustrated in Fig. 3-36B.

3-12. FANOUT CAPABILITY

Every logic device requires a specified current through its input terminals, and this current varies between the logical-0 and logical-1 states. At the logical-0 level, a specified *output* current must leave the input terminals. This is typically 1 to 2 milliamperes. At the logical-1 voltage level, *input* current is required, typically 30 to 80 microamperes.

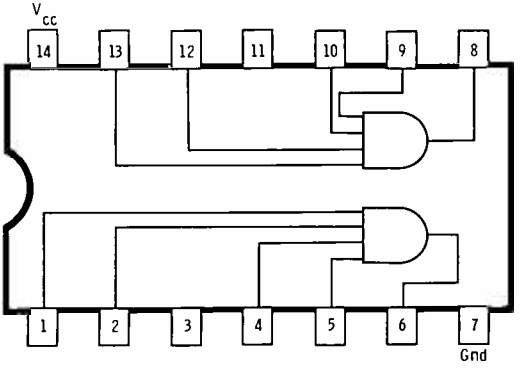


(*A*) *Functional diagram.*

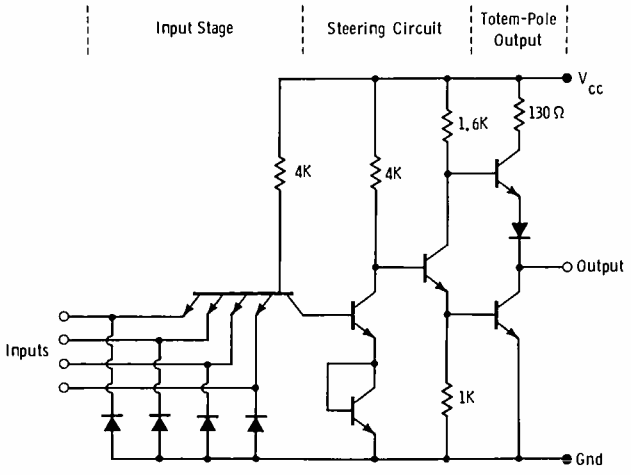


(*B*) *Circuit of one gate.*

Fig. 3-35. Quad two-input NAND gate.



(A) Functional diagram.



(B) Circuit of one gate.

Fig. 3-36. Dual four-input AND gate.

Fanout reflects the ability of an *output* terminal of a device to *sink* (absorb) current from a number (n) of loads at a logical-0 voltage level, and to supply current to a number of loads at a logical-1 voltage level. For example, a typical TTL device may be capable of sinking current from and supplying current to 10 loads ($n = 10$). A properly designed buffer gate may be capable of handling many more loads. As just one example, the Signetics buffer gate 54/7440 is specified as sinking current from or supplying current to 30 TTL loads ($n = 30$). Again, there is no substitute for the specification sheet in determining the fanout capability of any particular logic device.

3-13. HOW TO READ IC DESIGNATIONS

On the top of each integrated circuit, you will normally find three types of information:

1. The manufacturer's name, symbol, or initials
2. The manufacturer's date code
3. The type number of the device

The 7400 Series of devices in the TTL family is commonly used in modern digital systems. The following examples illustrate how to read the code:

1. SW7410N 7334: From the specification sheet, this is a Type 7410 triple three-input NAND gate, manufactured by Stewart Warner Corp. (SW). The date code gives the week of the year during which the device was manufactured. Thus, 7334 means the chip was manufactured during the 34th week of 1973.
2. 7512 MC7446: This is a 7446 BCD–to–Seven-Segment Decoder/Driver (from specification sheet), manufactured by Motorola (MC with imprinted map of Texas). The 7512 indicates the device was manufactured during the 12th week of 1975.
3. S7419 N7400 AH: This is a Type 7400 quad two-input NAND gate manufactured by Signetics Corp. during the 19th week of 1974.
4. ITT 7402J 7306: This is a 7402 quad two-input NOR gate manufactured by ITT during the 6th week of 1973.

EXERCISES

Q3-1. Read aloud the expressions: (A) AB, (B) $\overline{AB}$, (C) $\overline{A}\overline{B}$, (D) $A + B$, (E) $\overline{A + B}$, (F) $A \oplus B$.

Q3-2. Prepare the truth table for Fig. 3-37.

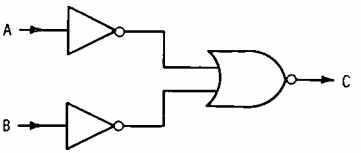


Fig. 3-37. Logic circuit for Q3-2.

Q3-3. Prepare the truth table for Fig. 3-38.

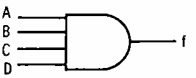


Fig. 3-38. Logic circuit for Q3-3.

Q3-4. (A) Prepare the truth table for the circuit of Fig. 3-39. (B) Draw a symbol for the equivalent logic function.

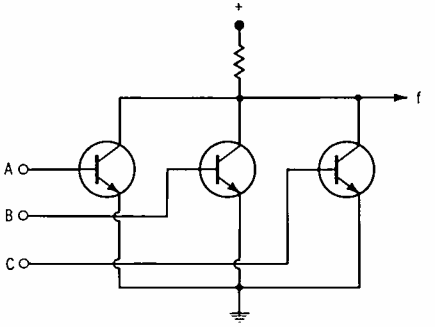


Fig. 3-39. Circuit for Q3-4.

Q3-5. Simplify the Boolean relationship $AC + ABC + A\overline{C}$ (A) by the algebraic method, (B) by the truth-table method.

Q3-6. See Fig. 3-40. Write the Boolean relationships for f1, f2, f3, f4, f5, f6, f7, and f8.

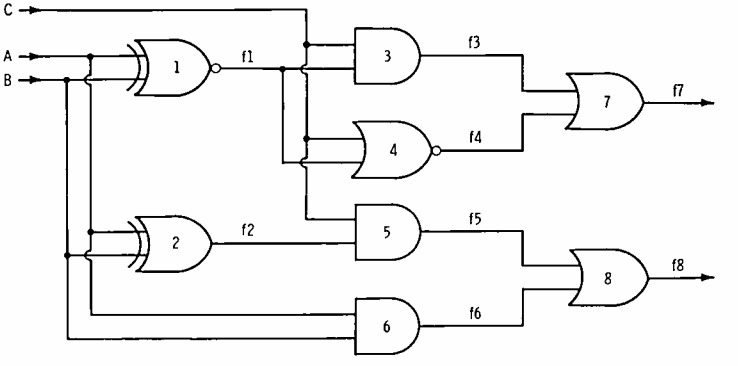
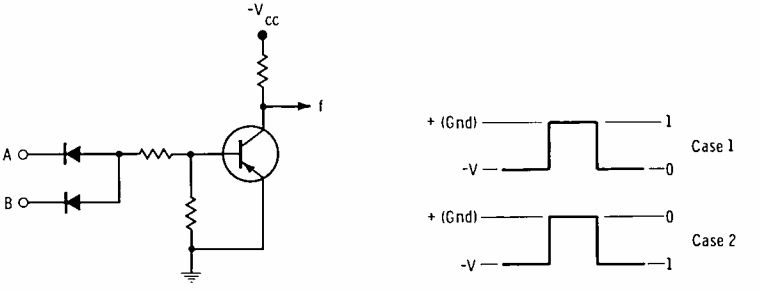


Fig. 3-40. Logic circuit for Q3-6.

Q3-7. See Fig. 3-41. Give function (f) for (A) case 1 inputs, (B) case 2 inputs.

Q3-8. Define *fanout.*



(A) *Circuit diagram.* (B) *Input waveforms.*

Fig. 3-41. Circuit for Q3-7.

CHAPTER 4

Logic Networks With Flip-Flops and Displays

Flip-flops are widely used whenever some form of storage or memory is required. The bistable flip-flop is basic to many logic networks. The most common types are the toggle (T), set-reset (RS), and JK. Flip-flops may be *clocked* (synchronous) or *unclocked* (asynchronous). Also basic to an understanding of logic networks are the study of *serial* and *parallel* systems and the use of *registers*.

The majority of digital integrated circuits are made up of AND gates, OR gates, inverters, and flip-flops. Of this group, flip-flops are the hardest to retain in your memory. We will therefore outline each individual type before proceeding to applications.

4-1. THE TOGGLE

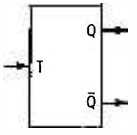
The triggered, or *toggle*, flip-flop requires only one input. All others require two or more inputs. The complementary outputs are normally designated Q and $\overline{Q}$ (read as "Q and Q not"). Sometimes the Q output is termed "1," and the $\overline{Q}$ is termed "0." If:

$$Q = 1, \text{ then } \overline{Q} = 0$$

$$Q = 0, \text{ then } \overline{Q} = 1$$

Fig. 4-1 illustrates the symbol and truth table for a toggle flip-flop. In the truth table, t_n is the time before the trigger pulse (present state), and t_{n+1} is the time after the trigger pulse.

There is no predetermined state for the two outputs when the circuit is first turned on. Thus, the state of the outputs following a trigger pulse cannot be predicted unless the present state is known. The circuit simply toggles (reverses) each time a trigger pulse arrives. Although simple, it is widely used.



(A) *Symbol.*

t_n		t_{n+1}	
Q	$\bar{Q}$	Q	$\bar{Q}$
0	1	1	0
1	0	0	1

(B) *Truth table.*

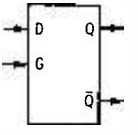
Fig. 4-1. Symbol and truth table for toggle flip-flop.

4-2. THE GATED-D TYPE

Fig. 4-2A shows the symbol for a *gated-D* flip-flop, termed a *latch circuit* after its corresponding relay symbol. Data to be transferred are applied to input D. The enabling gate (G) receives the enabling input. Transfer of data is blocked until the gate input is *asserted.* "Asserted" in this diagram means that data are transferred only when the gate goes from a low (0) level to a high (1) level. When G goes high, the data (either 1 or 0) are transferred to the Q output. Normally the gating voltage is related to the voltage level of the pulse, which can be a dc change. After the gate input threshold voltage has been exceeded, data input D is locked out.

The truth table for the gated-D flip-flop of Fig. 4-2A is shown in Fig. 4-2B. Note that the existing level on data input D is transferred to the Q output only when the gate voltage is a 1 (high). When G is unasserted (low, or 0), the outputs remain (latch) in their existing states (Q_n is the existing state, no change).

For example (Row 2), if the data input is still low when the next gate pulse arrives, there might be no change in the outputs. However, the circuit is *updated* because Q and $\bar{Q}$ have a specified value ($Q = 0$ and $\bar{Q} = 1$). Similarly (Row 4), if D is still high when G is again high, no change need occur, but the circuit is updated



(A) *Symbol.*

Row	D	G	Q	$\bar{Q}$
1	0	0	Q_n	$\bar{Q}_n$
2	0	1	0	1
3	1	0	Q_n	$\bar{Q}_n$
4	1	1	1	0

Q_n = Existing State (No Change)
G Unasserted = 0
G Asserted = 1

(B) *Truth table.*

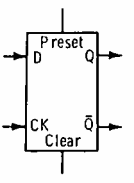
Fig. 4-2. Symbol and truth table for gated-D flip-flop.

($Q = 1$ and $\overline{Q} = 0$). When the gate is low (0), the outputs simply remain latched in their previous state, regardless of any change in the data input. The gate input may be clock pulses from a master timer.

Fig. 4-3 shows a D-latch with the addition of *direct set* (preset or set) and *direct clear* (sometimes termed *direct reset*) inputs. When the gate input is clocked, the clock signal is designated C, CK, CP, or CL. If preset and clear are both low (unasserted), data (D) are transferred to the Q output on the *positive* transition of the clock pulse. The preset and clear functions are independent of the clock pulse (overrides the clock). A high (1) on the preset (set) sends Q to 1 and $\overline{Q}$ to zero, regardless of the D or CK inputs. A high (1) on the clear (reset) input sends Q to 0 and $\overline{Q}$ to 1, regardless of the D or CK inputs. Data are normally synchronous with the clock pulse.

Fig. 4-3B illustrates the truth table for the circuit of Fig. 4-3A. Note that the preset and clear high (1) inputs override whatever other conditions exist, and that they cannot both be high simultaneously. The *asserted* value of preset or clear is *high* for the diagram of Fig. 4-3A.

Another D-latch circuit that is commonly encountered is presented in Fig. 4-4. Note the small circles on the clock, preset, and clear inputs. Remember that the small circle always indicates phase inversion. In this case, data (D) are transferred to the Q output only on the *negative-going* edge of the clock pulse. The asserted values of preset and clear are now zeros, which means that with a 1 (unasserted) on both preset and clear, data are transferred to the Q output on the negative-going edge of the CK pulse.



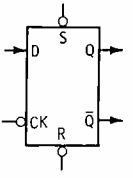
(A) *Symbol.*

Clock (CK)	D	S	R	Q	$\overline{Q}$
↑	0	0	0	0	1
↑	1	0	0	1	0
↓	x	0	0	Q_n	$\overline{Q}_n$
x	x	0	1	0	1
x	x	1	0	1	0
x	x	1	1	Not valid	

Q_n, $\overline{Q}_n$ = Present State, No Change
S = Direct Set or Preset
R = Direct Reset or Clear
x = Don't-Care Case (Can Be Either 0 or 1)

(B) *Truth table.*

Fig. 4-3. Symbol and truth table for D-latch with preset and clear.



(*A*) *Symbol.*

CK	D	S	R	Q	$\bar{Q}$
↘	1	1	1	1	0
↘	0	1	1	0	1
↗	x	1	1	Q_n	$\bar{Q}_n$
x	x	1	0	0	1
x	x	0	1	1	0
x	x	0	0	Not Valid*	

S = Direct Set (Preset)
R = Direct Reset (Clear)
S or R Asserted = 0
S or R Unasserted = 1
Q_n, $\bar{Q}_n$ = State at t_n (No Change)
x = Don't-Care Case
* Both outputs will temporarily be in 1 state, but will not remain after transition from unasserted to asserted values.

(*B*) *Truth table.*

Fig. 4-4. Symbol and truth table for D-latch with inverted CK, R, and S.

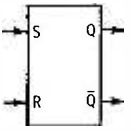
IMPORTANT: In many cases, the CK input of Fig. 4-4A is drawn without the small circle. In this case, the same conditions as above prevail, but data are transferred to the Q output on the positive-going edge of the CK pulse.

4-3. THE SET-RESET FLIP-FLOP

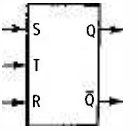
There are three basic types of *set-reset* (commonly termed *RS*) flip-flops. Fig. 4-5A shows the symbol for the straight RS flip-flop. The symbol for a toggle RS flip-flop is shown in Fig. 4-5B, and that for a clocked RS flip-flop is shown in Fig. 4-5C. In the clocked version, the transfer occurs only when the clock pulse changes from 0 to 1.

Outputs are drawn directly opposite the inputs to which they correspond. A 1 on the set (S) input (while $R = 0$) sets the Q output to 1; therefore $\overline{Q} = 0$. A 1 on the reset (R) input sets $\overline{Q}$ to 1; hence $Q = 0$. Thus, for the clocked flip-flop (Fig. 4-5C), you are able to introduce data at a given time (t_n) and have the data transferred at a later time (t_{n+1}) determined by the clock-pulse timing. This is in common with the D-latch described above.

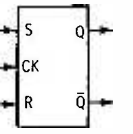
The truth table for the toggle RS flip-flop of Fig. 4-5B is shown in Fig. 4-5D. Note that for any RS flip-flop, R and S must not be high at the same time. If both R and S are held low for trigger application, the flip-flop will reverse states (toggle).



(*A*) *Straight.*



(*B*) *Toggle.*



(*C*) *Clocked.*

T	S	R	Q	$\bar{Q}$	Time	Condition
0	0	1	1	0	t_n	Reset
0	0	0	0	1	t_{n+1}	
0	1	0	0	1	t_n	Set
0	0	0	1	0	t_{n+1}	
1	0	0	1	0	t_n	Toggle
0	0	0	0	1	t_{n+1}	
1	0	0	0	1	t_n	Toggle
0	0	0	1	0	t_{n+1}	
0	1	0	1	0	t_n	Set
0	0	0	1	0	t_{n+1}	
0	0	1	0	1	t_n	Reset
0	0	0	0	1	t_{n+1}	
0	1	1	Not Allowed (Indeterminate)			
1	1	1				

t_n = Time Immediately Preceding Input Pulse
t_{n+1} = Time Immediately Following Input Pulse

(*D*) *Truth table* (*toggle*).

Inputs at t_n		Outputs at t_{n+1}	
R	S	Q	$\bar{Q}$
0	0	No Change	
0	1	1	0
1	0	0	1
1	1	Not Allowed (Indeterminate)	

t_n = Time Just Preceding CK
t_{n+1} = Time Just After CK

(*E*) *Truth table* (*clocked*).

Fig. 4-5. Three types of RS flip-flops.

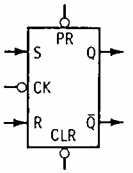
Fig. 4-5E is the truth table for the clocked RS flip-flop of Fig. 4-5C. Data on R or S are transferred to the output only on (or immediately following) the positive clock transition. If both R and S are held low, no change occurs for any application of clock pulses.

In Fig. 4-6, the functions of direct reset and direct clear are added to the RS clocked flip-flop. Note the small circles on the CK, PR, and CLR inputs. The direct preset (set) and clear (reset) inputs override all other conditions when asserted. In this diagram, data are transferred from R and S on the negative-going edge of the CK pulse, provided that PR and CLR are both unasserted (1).

If PR is asserted (0) with CLR = 1, output Q is set to 1 ($\overline{Q} = 0$), regardless of the states of CK, R, and S. Conversely, if CLR is asserted (0) with PR = 1 (unasserted), output Q is cleared to 0 ($\overline{Q} = 1$).

The last row of the truth table in Fig. 4-6B refers to the text. The following rules apply to this row:

1. If preset and clear are both asserted (0 in this case), both Q and $\overline{Q}$ become high (1) temporarily, but this condition will not persist.



(A) *Symbol.*

Inputs at t_n				Outputs at t_{n+1}	
R	S	PR	CLR	Q	$\bar{Q}$
0	0	1	1	Q_n	$\bar{Q}_n$
0	1	1	1	1	0
1	0	1	1	0	1
1	1	1	1	Not Allowed (Indeterminate)	
x	x	0	1	0	1
x	x	1	0	1	0
x	x	0	0	See Text	

t_n = Time Immediately Prior to Negative Edge of CK
t_{n+1} = Time Immediately After Negative Edge of CK
Q_n, $\bar{Q}_n$ = State at t_n
PR and CLR Asserted = 0
PR and CLR Unasserted = 1
x = Don't-Care Case

(B) *Truth table.*

Fig. 4-6. RS flip-flop with direct preset (set) and clear (reset).

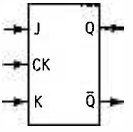
2. When preset is unasserted after clear has been unasserted, Q will remain *high* until a clock pulse arrives.
3. When clear is unasserted after preset has been unasserted, Q will remain *low* until a clock pulse arrives.

4-4. THE JK FLIP-FLOP

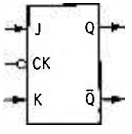
The JK flip-flop (Fig. 4-7) is identical to the RS flip-flop except for the one condition mentioned below. The following similarities exist:

1. The Q output is drawn opposite the S input on the RS symbol and opposite the J input on the JK symbol. This means that a logical 1 on the S or J input causes the Q output to go to logical 1 at the significant edge of the clock pulse ($\overline{Q} = 0$).
2. The $\overline{Q}$ output is drawn opposite the R input on the RS symbol and opposite the K input on the JK symbol. This means that a logical 1 on the R or K input causes the $\overline{Q}$ output to go to logical 1 at the significant edge of the clock pulse ($Q = 0$).

The only difference is in the condition where both inputs are high (logical 1) when CK occurs. In the RS toggle flip-flop, if R and S are both high when clocked, the outputs are indeterminate (unpredictable). In the JK flip-flop, if both J and K are high when clocked,



(*A*) *Positive CK trigger.*



(*B*) *Negative CK trigger.*

Inputs at t_n		Outputs at t_{n+1}	
J	K	Q	$\bar{Q}$
0	0	Q_n	$\bar{Q}_n$
0	1	0	1
1	0	1	0
1	1	Complement*	

*Complement means that the outputs reverse (toggle) from their existing states when clocked.

(*C*) *Truth table.*

Fig. 4-7. Symbols and truth table for JK flip-flops.

both output levels switch (toggle) from their existing states (Fig. 4-7C). Thus, there are no indeterminate output states for the JK flip-flop. When direct preset and/or clear are used, the same conditions as noted for Fig. 4-6B exist, except for the one condition noted above.

NOTE: The JK flip-flop can also be used as a "toggle" (complement of preexisting level) by tying inputs J and K together and applying the pulse to the common input.

4-5. MASTER-SLAVE OPERATION

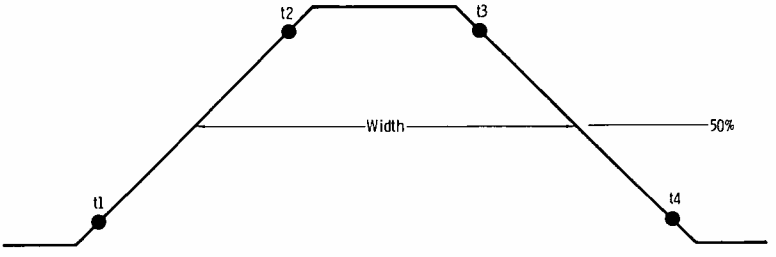
Either data-latch, RS, or JK flip-flops may constitute a *master-slave* type of operation. The JK type is most commonly used. Master-slave means that data *stored* in the master section are transferred to the slave section at a later time. Both the leading and trailing edges of a clock pulse have a particular function in this mode of operation. There are two basic forms of master-slave arrangements:

1. Pulse-triggered master-slave
2. Edge-triggered master-slave.

The Pulse-Triggered Master-Slave Flip-Flop

The function of a pulse-triggered master-slave flip-flop is shown in Fig. 4-8. With this type, the logical state of the inputs should not be allowed to change when the clock pulse is in the high state. Note that the input data are not disabled until t3. Therefore, the data immediately preceding t1 are the desired information to store at t2. The transfer of data to the slave (output) section occurs at t4.

If the data inputs should change during the time between t1 and t3, the pulse-triggered master-slave flip-flop will usually store the



t1: Isolates slave from master.
t2: Enters information on data inputs to master.
t3: Disables data inputs.
t4: Transfers information from master to slave.

Fig. 4-8. Action of pulse-triggered master-slave flip-flop.

high level, whether the high level occurs first or last. This is termed "ones catching" and is not a conventional function of the pulse-triggered master-slave flip-flop.

The pulse width is taken between the 50%-amplitude points of the clock waveform. For example, if the amplitude of the clock pulse is 3 volts, the pulse width is measured between the 1.5-volt levels of the leading and trailing edges. The rise time of the edges is measured between the 10%- and 90%-amplitude points.

The Edge-Triggered Master-Slave Flip-Flop

With the edge-triggered master-slave flip-flop (Fig. 4-9), the data inputs are disabled (locked out) until the clock pulse goes *high*. When this occurs, the data inputs are enabled, and the data level is

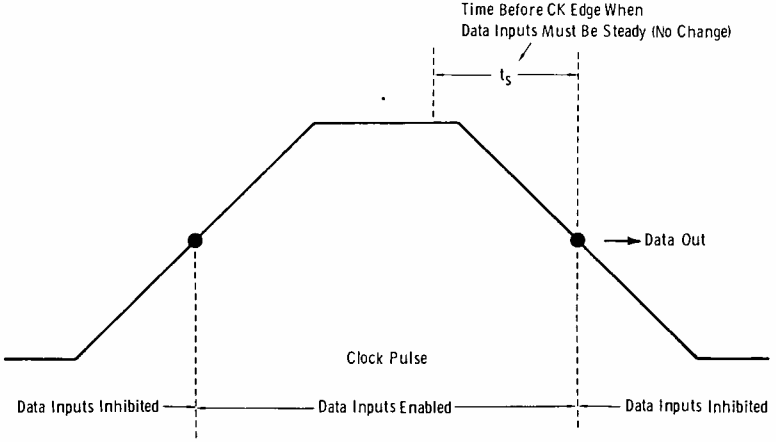


Fig. 4-9. Action of edge-triggered master-slave flip-flop.

stored in the master. If the data input level changes during the clock-pulse duration, the *most recent change* (prior to the minimum t setup shown) will be stored. If a change occurs during interval t_s, the most recent change will be stored and could be inaccurate. The t_s interval should normally see no change in the data level.

4-6. THE BINARY COUNTER

The following analysis refers to Fig. 4-10. Read from right to left so that the least significant digit (LSD) is on the right for convenience. In this drawing, the Q output is indicated as 1, and the $\overline{Q}$ output is indicated as 0.

Sixteen trigger pulses are shown feeding the toggle (T) input of FF1. When the flip-flop is in the set (S) state, the 1 output reads 1 (high level), and the 0 output reads 0 (low level). If the circuit is in the clear (C), or reset, state, the 1 output reads 0, and the 0 output reads 1. Further assume that this series of flip-flops can be reset (cleared) so that all 1 outputs are 0 simultaneously. After this, when the clear (C) side changes from the 0 state to the 1 state (1 to 0 on the set side), a pulse is sent to the input of the flip-flop of the next higher order. The circuit action is as follows:

Prior to pulse 1: All four flip-flops are in the clear state; each output reads 0, and the total counter output is 0000.

Pulse 1: Changes FF1 to the set state, producing a 1 at the set output. However, neither FF2 nor any of the following stages

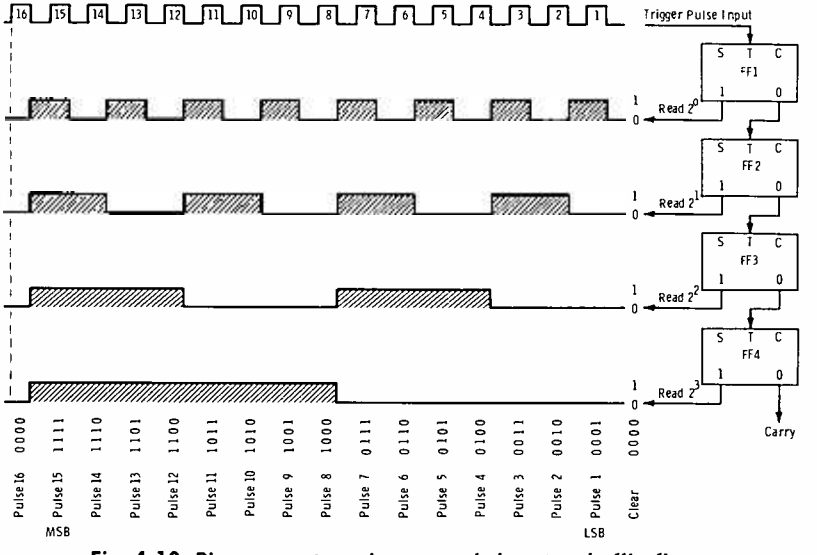


Fig. 4-10. Binary counter using set and clear toggle flip-flops.

receives a signal because FF1 produces a pulse at its clear (C) output only when changing from the set state to the clear state. Note that the counter output is now 0001.

Pulse 2: Changes FF1 from set to clear (reset), producing a 0 at the set output and a 1 at the clear output. The C output pulse changes FF2 from the clear to the set state (the S output of FF2 now reads 1). The clear output of FF2 is 0, and the remaining flip-flops are not affected. The total counter output is now 0010, or the binary equivalent of decimal 2.

Pulse 3: Changes FF1 to the set state again, but the following flip-flops remain in the previous state. The output is now 0011.

Pulse 4: Changes FF1 and FF2 to the clear state, and FF3 to the set state. We now have 0100.

If you continue this analysis, you find that the circuit will count up to 15 pulses and then return to clear (0000) on the sixteenth pulse. The highest possible binary number is 1111, or 15. The modulus of this counter is 15 + 1 = 16. In binary form:

$$\begin{array}{rl} & 1111\ (15) \\ + & \underline{0001}\ (\ 1) \\ = & \underline{1}0000\ (16) \end{array}$$

This tells you that on the sixteenth pulse, a 1 exists (underlined above) at the carry output of FF4 so that an additional FF5 would count this pulse. Otherwise, the counter is simply in the clear condition and ready to repeat counting on the seventeenth pulse.

Binary Addition by Counting

The status of flip-flops is often shown by indicator lights on an operating panel. For the 1 state, an indicator bulb lights. In Fig. 4-11, seven pulses are initially applied to the T input of FF1. These seven bits would create a binary count of 0111 or 4 + 2 + 1 = 7

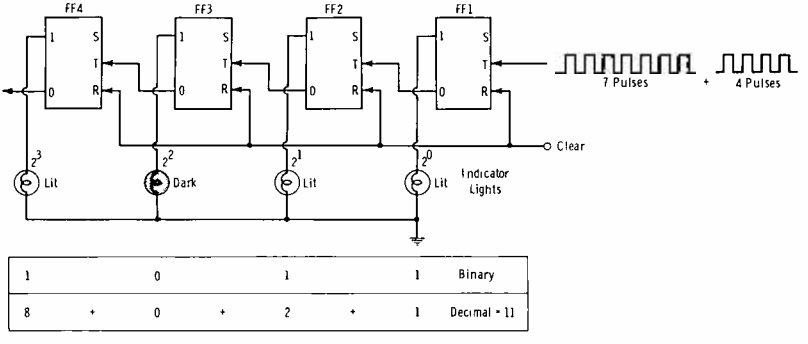


Fig. 4-11. Principle of binary addition by counting.

decimal. The addition of four more bits will result in a final count of 1011 as shown. This is equivalent to decimal $8 + 2 + 1 = 11$. Thus, the result of entering seven pulses and then four more pulses is $7 + 4 = 11$.

Details of indicating circuitry are developed later in this chapter.

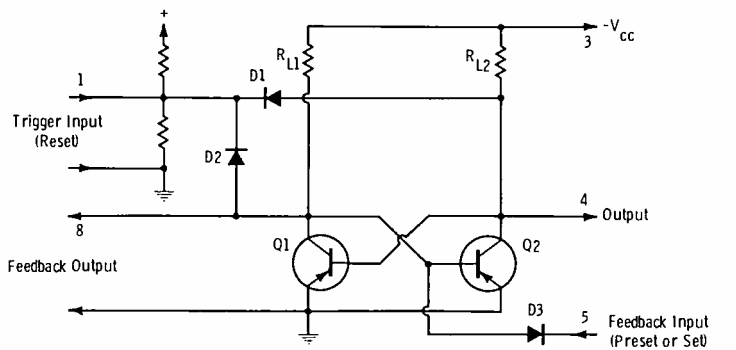
Binary Division by Uneven Numbers

For binary division in calculators, a shift register (Section 4-9) may be used. For a fixed uneven count, such as must be used in synchronizing generators, a special arrangement of flip-flops is employed. This arrangement may be called a *counter* or *divider;* either term is correct.

Although digital counting circuitry normally employs ICs, it is helpful to use a discrete-component diagram such as Fig. 4-12 to clarify the circuit function. In Fig. 4-12, D1 and D2 provide negative-pulse steering for binary (flip-flop) Q1-Q2. A negative trigger-pulse input at point 1 turns on whichever transistor is off; it has no effect on the transistor already on, since that base is already negative. The arrival of the next negative trigger again reverses (resets) the state of the binary. However, this action occurs *only* if there is no conflicting information from D3, that is, if D3 is an open switch. Thus, if D3 has been opened, the binary has been *preset* or *set* to operate as an ordinary binary (divide-by-two) multivibrator. The output voltage at point 4 is close to zero when Q2 is saturated, and it is $-V_{CC}$ when Q2 is cut off.

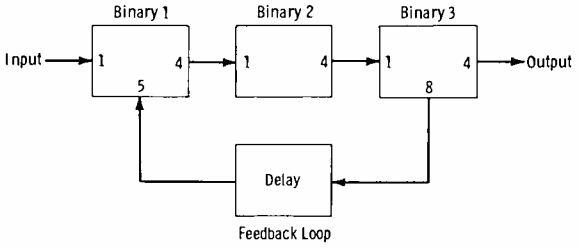
In Fig. 4-13A, three of the binaries of Fig. 4-12 are interconnected to form a chain. Normally (without the feedback loop), we would have 2^3, or 8-to-1, division. Now see how the action is modified by feedback from the third binary to the first one.

In the waveforms of Fig. 4-13B, the output (point 4) of binary 3 is in a 0 state at the extreme left of the drawing. Therefore, Q2 of

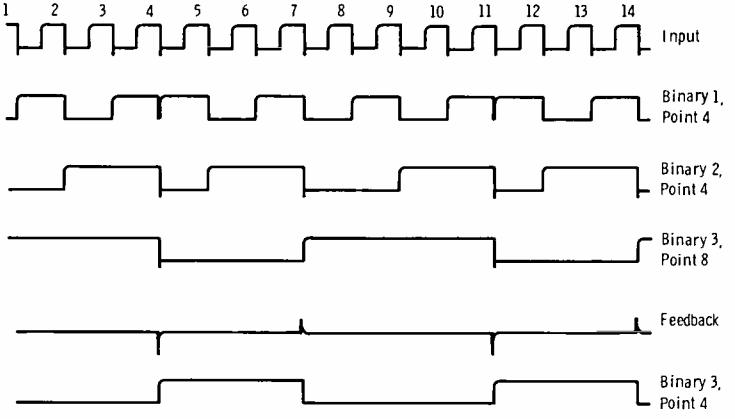


Note: Pin Numbers 1, 4, 5, and 8 are correlated with blocks of Fig. 4-13.

Fig. 4-12. Binary counter stage with preset or set input.



(*A*) *Diagram.*



(*B*) *Waveforms.*

Fig. 4-13. Divide-by-seven counting chain.

that binary is cut off (open switch) so that the output voltage is $-V_{CC}$. Also, Q1 of that binary is saturated, so point 8 is at ground. Since point 8 is connected back to D3 (point 5) of the first binary, this D3 is open (no forward bias), and binary 1 is counting.

After four input pulses to the first binary, Q1 of the third binary is changed from saturation to cutoff, and point 8 goes negative. This voltage is fed back (through a slight delay so as not to disturb the triggering of the second binary) to point 5 of binary 1. Now D3 of the first stage is forward-biased and immediately brings the base of Q2 to a negative value. So Q2, which was driven to cutoff at the trailing edge of the pulse shown as number 4 in the input waveform, is immediately set back to a 1 state. The resulting pulse serves as the *first* of the next group of four pulses applied to binary 2. The result is that after only seven pulses from the input triggering signal, the last counter again triggers, setting the first binary to repeat the cycle. This is most evident in the waveform at point 4 of binary 1; note the "set" occurring at the trailing edges of input pulses 4 and

11 (seven-pulse interval). Thus, this group of three binary scalers produces a count of 7 to 1 rather than the normal 8 to 1 obtained when no feedback loop is employed.

When scalers are used as counters in sync generators, the method just described is the fundamental principle on which they operate. The action of a chain of binary counters of any given number of units, with a feedback loop from the last unit to any preceding one in the chain, has the following relationship:

$$\text{Total division} = 2^n - 2^p$$

where,

n is the total number of binary scalers,

p is one less than the order number of the scaler to which feedback from the output scaler is applied.

If the feedback pulse is applied to the input of the first scaler, $p = 0$; if it is applied to the input of the second scaler (output of first scaler), $p = 1$; if it is applied to the input of the third scaler, $p = 2$; etc. Thus for Fig. 4-13:

$$2^n - 2^p = 2^3 - 2^0 = 8 - 1 = 7$$

4-7. THE DECADE COUNTER

A large number of applications require a counter operating in the *decade* mode rather than a purely binary mode. To do this, it is necessary for the four flip-flops to count to 9 (1001) and then reset to 0 on the tenth pulse. (Review binary-coded-decimal notation, Section 2-9.) The tenth pulse then carries 1 to the first flip-flop of the following decade.

Fig. 4-14 shows one common method of doing this. On the tenth count (1010), FF1 (2^1) and FF3 (2^3) are in the 1 state. These outputs are fed to a two-input AND gate, the output of which clears (resets) the counter to 0.

There are several forms of decade counters, but all operate on the principle of resetting the first group of four flip-flops (first decade, 0 to 9) to zero after nine pulses, and carrying 1 to the next decade group. A decade counter may be contained in a single IC chip.

4-8. SERIAL AND PARALLEL OPERATION

In Fig. 4-15A, binary group 1011 is sent as a pulse series on one wire. In Fig. 4-15B, each bit of 1011 is sent on an individual wire, and all bits may be sent simultaneously. This illustrates the basic difference between the serial and parallel modes of operation.

In the binary counter of Fig. 4-10, the input was a series of 16 pulses, and there were four parallel output feeds. You can say this

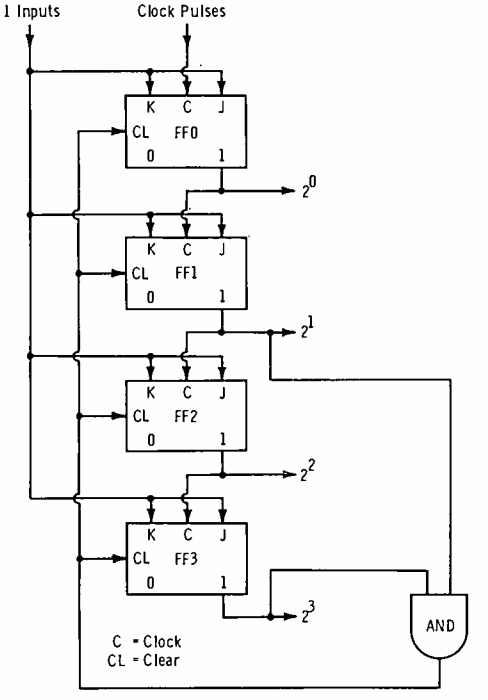


Fig. 4-14. One type of decade counter.

is one form of serial-to-parallel converter. In practice, you will find any of the following:

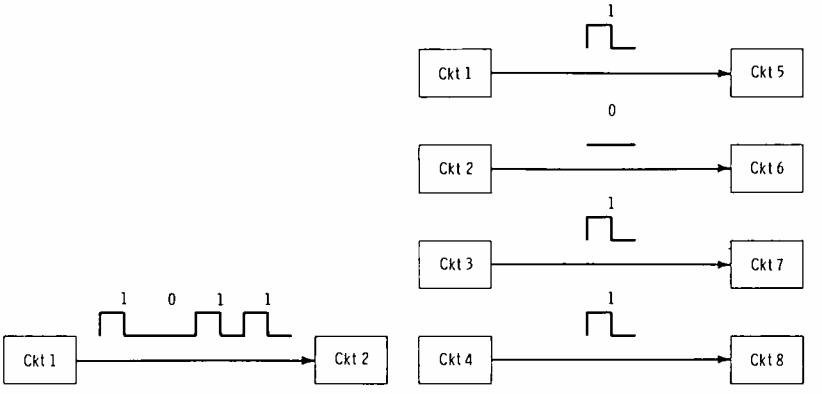
1. Serial in, serial out
2. Serial in, parallel out
3. Parallel in, serial out
4. Parallel in, parallel out

4-9. SHIFT REGISTERS

A shift register is simply a series of flip-flops (usually within one IC) employing delay elements and a reset clock pulse. A clocked clearing pulse applied to the reset line causes a binary number to be shifted through the stages instead of being "cleared." Fig. 4-16A shows a left-shift register (basic multiplier), and Fig. 4-16B shows a right-shift register, or basic divider. Review Sections 2-7 and 2-8 for binary multiplication and division.

The operation of the circuit in Fig. 4-16A is as follows:

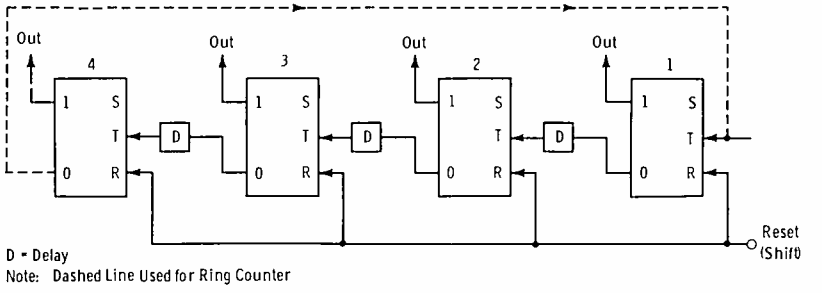
1. Assume FF1 has been triggered to the 1 state. All other stages are in the 0 state. A 1 exists at the 1 output of FF1.



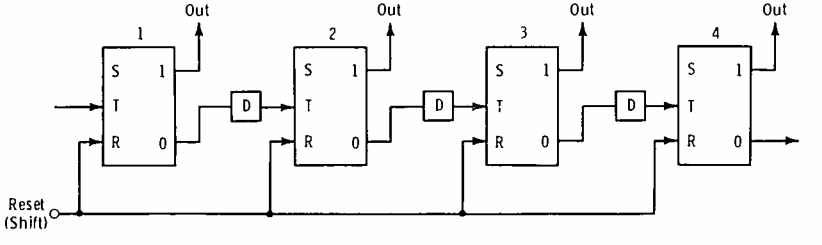
(*A*) *Serial* (*one wire*). (*B*) *Parallel* (*four wires*).

Fig. 4-15. Serial and parallel operation.

2. A reset (clear) pulse is applied to the reset line. This will reset FF1 (1 output goes to 0 level). The 0-to-1 change of the reset (0) output is applied to the trigger input of FF2 through a delay equal to at least the reset-pulse duration, thus triggering FF2 to the 1 state after the clearing pulse has ended. The output of FF1 is now 0, and the 1 has been shifted to the out-



(*A*) *Left-shift register.*



(*B*) *Right-shift register.*

Fig. 4-16. Principle of shift registers.

put of FF2. None of the remaining flip-flops is affected since the 0 outputs are already at the high level (1 outputs are 0).

3. Another clearing pulse is entered on the reset line. As a result, FF2 is cleared and sends a trigger to FF3, setting the output of FF3 to 1. Successive clearing pulses progressively shift the binary digit to the left. Shifting the binary digit through the last flip-flop clears the register until the next trigger occurs at FF1.

When the reset output of the last flip-flop is routed back to the input as shown by the dashed line, repeated entry of clearing pulses

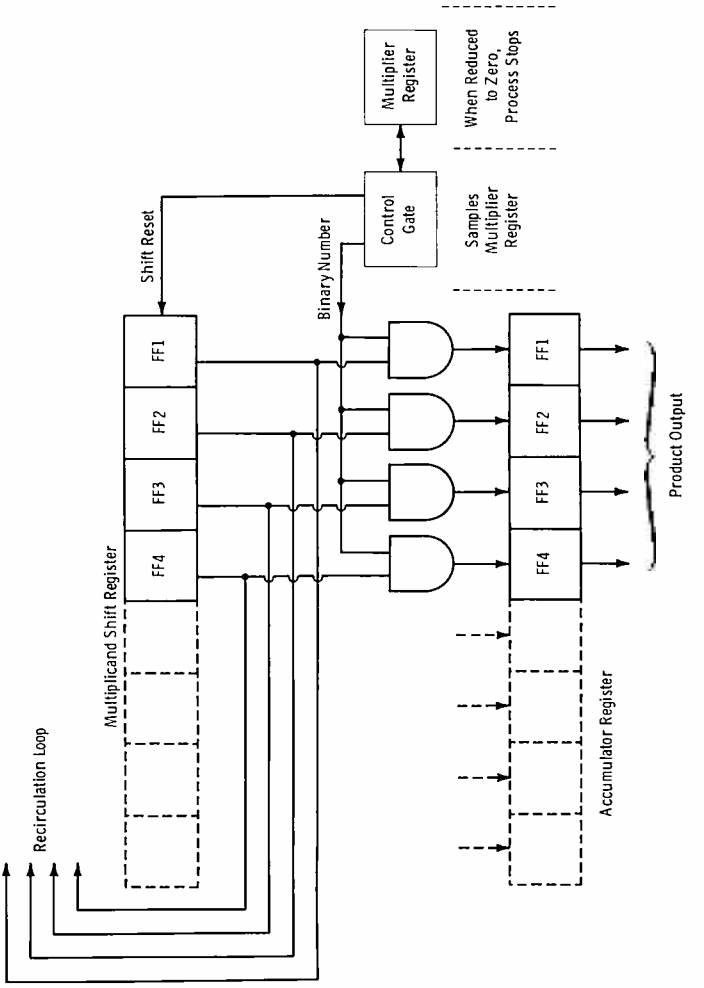


Fig. 4-17. Basic binary multiplication by shifting.

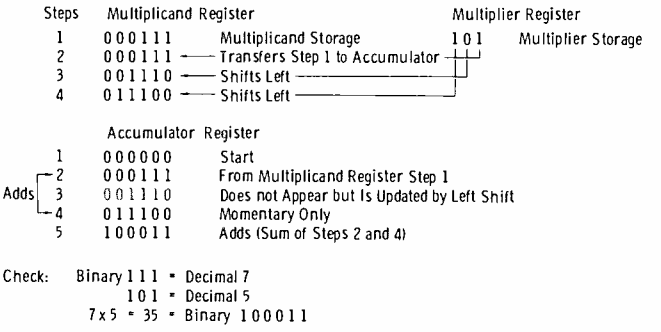


Fig. 4-18. Action of multiplier in Fig. 4-17.

causes the numbers to be circulated continuously within the register. This arrangement is termed a *ring counter.*

The arrangement of registers for basic binary multiplication is shown in Fig. 4-17. The binary multiplier is placed in a temporary storage register where it can be sampled by a control-gate system. The multiplicand is placed in a shift register, which drives the accumulator register through a series of AND gates as shown.

See Fig. 4-18 and assume that binary 111 is to be multiplied by binary 101. The operation takes place in the following steps:

Step 1: Binary 101 is placed into the multiplier register, and binary 111 is placed into the multiplicand register. The accumulator register starts with 000.

Step 2: The LSB is sampled by the control gate. Since this number is a 1, coincidence occurs with the number 111 stored in the multiplicand register, transferring 111 to the accumulator.

Step 3: A clear (shift) pulse is sent at the second bit. Since this is a 0, no coincidence occurs at the AND gates, and this partial product is not transferred to the accumulator. It is simply shifted left in both registers. This is termed *updating* the register.

Step 4: At the next shift pulse, since the bit is a 1, coincidence occurs at the AND gates, and the product in the accumulator becomes 11100. Since the multiplier register is now reduced to 0, Steps 2 and 4 add in the accumulator (Step 5), and action stops.

Step 5: The sum of Steps 2 and 4 appears on the readout in the accumulator. The momentary result of Step 4 (11100) is added to the 111 already in the accumulator, resulting in the final product, 100011. Note that the accumulator register must have twice the capacity of the multiplier register.

When the shift register is cleared for any reason such as temporary readout, the number is routed back through the recirculation loop and loaded back into the register.

Remember that one shift to the left multiplies the number by two. A shift to the right divides the number by two. The arrangement of Fig. 4-16B is used in similar circuitry for binary division.

Shift registers are used in numerous applications, as will be evident in the following chapters. Applications are not confined to arithmetical functions.

4-10. THE MATRIX AND ENCODING-DECODING

An *encoder* is a device capable of translating from one method of expression to another. A *decoder* is a device that determines the meaning of a set of signals and initiates logical operations based on

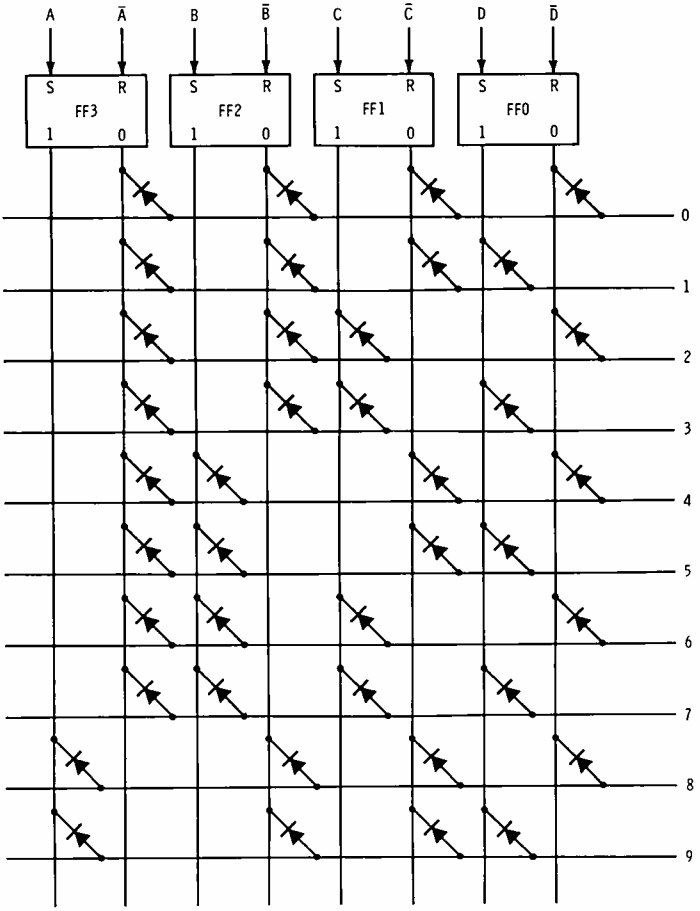


Fig. 4-19. Diode matrix decoder.

those signals. A *matrix* is a set of switching elements that selects one or more output channels according to the combination of input signals present.

A diode matrix decoder is illustrated in Fig. 4-19. This matrix provides a means of sensing the base-10 equivalent of a binary number stored in the flip-flops. This is a bcd count, previously described in connection with Fig. 4-14. A bcd counter is normally contained in a single IC chip.

For the first flip-flop stage, FF0 (for 2^0), the diodes are connected alternately from the 0 and 1 outputs to the horizontal lines. For the second stage, FF1 (for 2^1), the diodes are connected in pairs. For the third stage, FF2 (for 2^2), they are in groups of four. For the fourth stage, FF3 (for 2^3), they are in groups of eight. This diode matrix arrangement results in the base-2 place values of 8-4-2-1 as required.

The diode matrix decoder action is more clearly revealed by Fig. 4-20. Transistors Q0, Q1, Q2, and Q3 have a small positive voltage

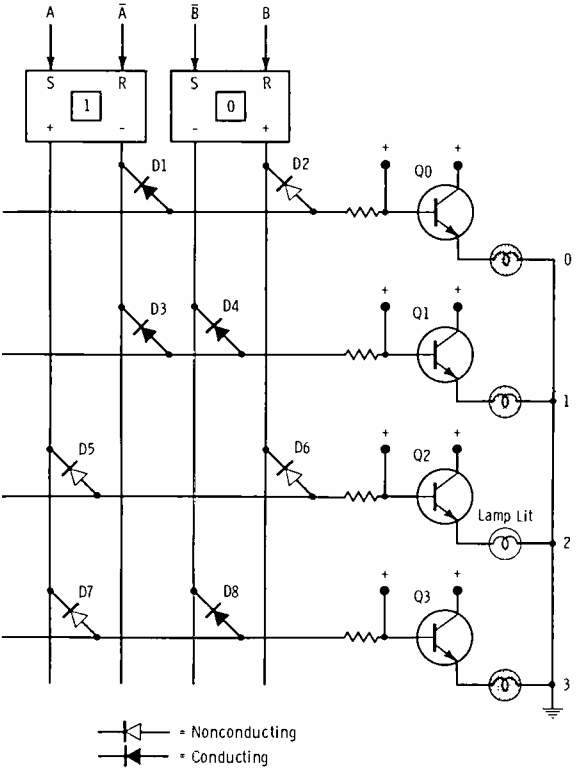


Fig. 4-20. Decode action for converting binary 10 to decimal 2.

applied to the base, saturating the npn transistors. When not connected to the matrix, the transistors are closed switches, lighting the lamps.

Fig. 4-20 shows the condition for $A\overline{B}$, or binary 10, or decimal 2. Note that the 0 outputs are designated negative polarity, and the 1 outputs are designated positive polarity. Thus diodes D1, D3, D4, and D8 are conducting (closed switches), while all other diodes are reverse-biased (open switches).

The resulting negative potential applied to the bases of Q0, Q1, and Q3 overrides the small forward bias and cuts off those transistors. Only Q2 is conducting, closing that switch and applying power to the lamp indicating decimal 2. Two binary digits have just four (2^2) combinations, for a count from 0 to 3. For any number appearing in the decade counter, only one driver transistor will conduct to light the associated lamp.

Fig. 4-21 shows a simple encoder using OR gates for decimal-to-binary conversion. In this example, decimal 6 is converted to the equivalent binary 0110. The input data could be from a keyboard, punched card, or tape (Chapter 6).

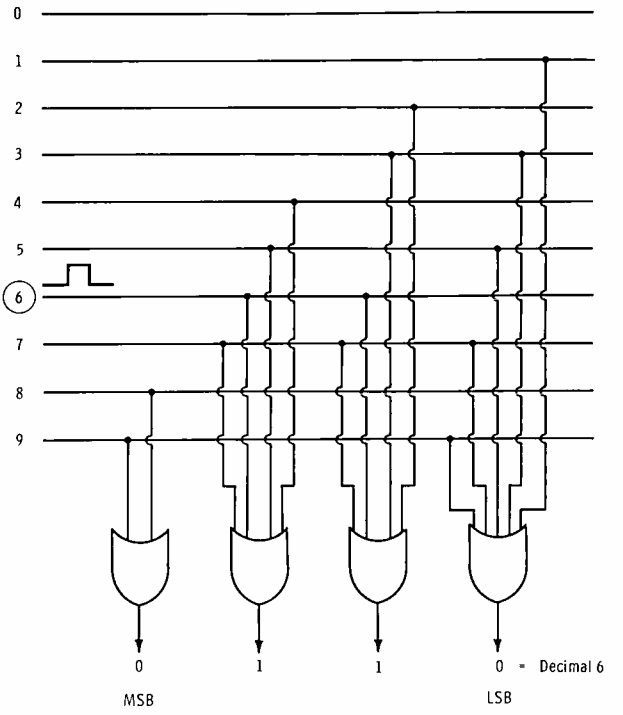


Fig. 4-21. Encoder for decimal-to-binary conversion.

4-11. INDICATING DEVICES AND CIRCUITRY

Numerous types of readout devices exist, all with their particular advantages and disadvantages. The most common general categories are:

1. Tube-type displays with individual cathodes shaped in the form of numbers or letters of the alphabet. A popular example is the Nixie tube manufactured by the Burroughs Corporation.
2. Tube-type displays with segmented numbers or characters. Each segment has an individual cathode. A popular example is the RCA Numitron.
3. The segmented light-emitting-diode (LED).
4. The LED matrix consisting of individual LEDs that are illuminated by appropriate logic circuitry to form numbers, letters of the alphabet, or various symbols.

These types are described in the following subsections.

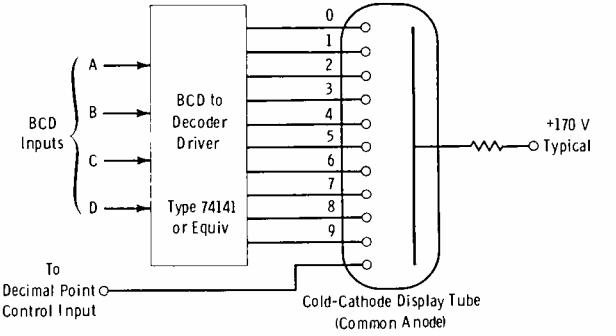
Tubes With Individual Cathodes

Fig. 4-22A shows the principle of a Nixie-type display tube. A negative voltage applied to a selected cathode causes gas ionization around that cathode, lighting the selected number. The corresponding truth table is shown in Fig. 4-22B. Note that only one cathode is energized at a time for the selected number. Fig. 4-22C is a photograph showing a typical display; the number 3.2 is displayed on two tubes. Each tube requires an individual decoder/driver. The characters may be up to 2 inches high, are brightly illuminated, and may be seen at a distance.

Segmented Tube Displays

The standard configuration for most seven-segment display devices is shown in Fig. 4-23A. In some types, the numbers are slanted at an angle of 2° to 10° from the vertical. Fig. 4-23B illustrates how each segment is driven from the decoder/driver IC. Fig. 4-23C shows a practical application for a *floating decimal* arrangement. The decimal-point input (DP_I) receives the control voltage from a separate control source; DP_O is the decimal-point output to the display tube. If each display device contains a decimal-point element, the decimal point may be placed at any required position in the number sequence.

The terminal designated L/T in Fig. 4-23C is a light-test provision. As can be observed from Fig. 4-23A, the number 8 requires that all seven segments be illuminated. When terminal L/T is grounded, all seven segments are lit, and the number 8 is displayed. This provides a quick test to assure that a single segment is not burned out. With multiple display devices, the L/T terminals are



(*A*) *Driver arrangement.*

Input				Output On*
D	C	B	A	
L	L	L	L	0
L	L	L	H	1
L	L	H	L	2
L	L	H	H	3
L	H	L	L	4
L	H	L	H	5
L	H	H	L	6
L	H	H	H	7
H	L	L	L	8
H	L	L	H	9
H	L	H	L	None
H	L	H	H	None
H	H	L	L	None
H	H	L	H	None
H	H	H	L	None
H	H	H	H	None

H = High Level, L = Low Level
*All other outputs are off.

(*B*) *Truth table.*

(*C*) *Example of display.*

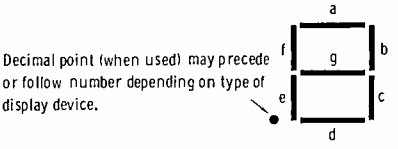
Fig. 4-22. Display tubes with individual cathodes.

all tied together through a common grounding test switch so that all displays are checked simultaneously.

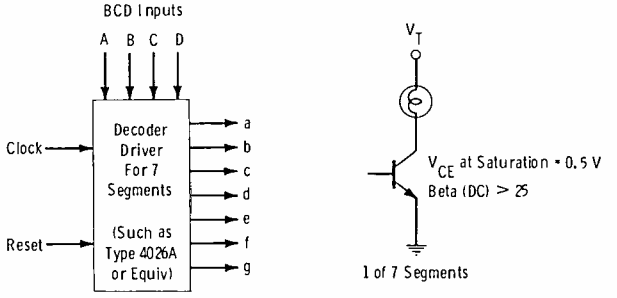
Note that for a seven-segment device, the decoder-driver energizes as many segments simultaneously as are required for the given character. This contrasts with the Nixie driver logic, which has only one output per character.

A *fixed decimal* application is shown in Fig. 4-24. In this case, only one display device contains the decimal point.

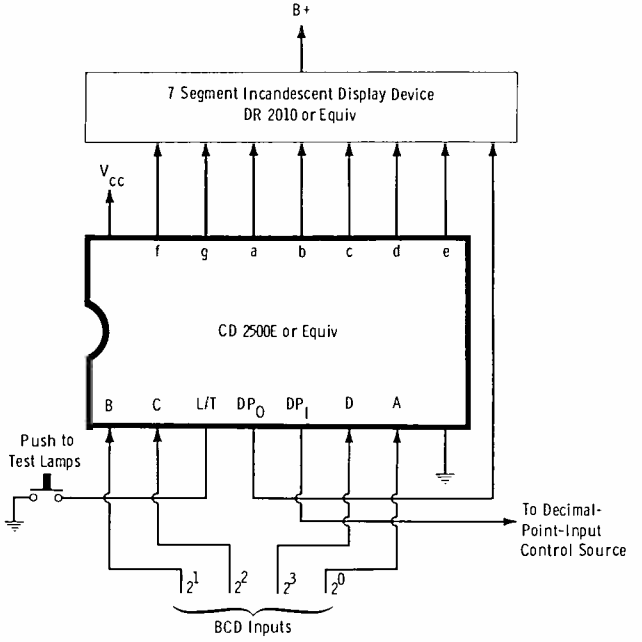
The seven-segment incandescent display tube is typified by the RCA Numitron series illustrated with physical dimensions in Fig. 4-25A. This is a nine-pin glass tube with seven thin lamp filaments



(A) Standard segment configuration.



(B) Method of driving segments.



(C) Typical application.

Fig. 4-23. Principles of seven-segment display.

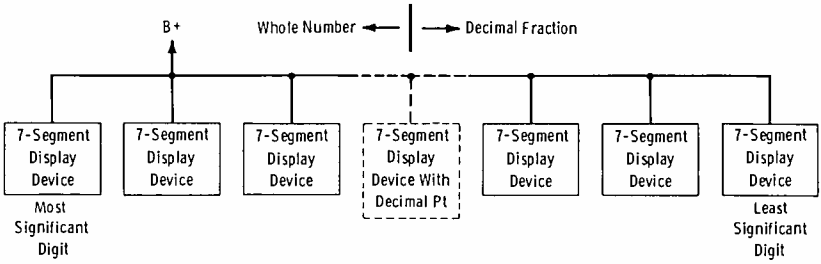


Fig. 4-24. Seven-digit seven-segment display with fixed decimal point.

(one for each segment) viewed from the side. The voltage required is 3.5 to 5 volts, and the tube can be interfaced directly with standard logic circuitry. However, the current required is about 24 mA per segment. As with any incandescent lamp filament, when the display is first turned on, the surge current can approach 300 mA per segment. The number 8 (seven segments) would require a surge current of close to 7×300 mA $= 2.1$ amperes, and the driver IC must be appropriately rated. The surge duration is only a few milliseconds.

The characteristics and physical dimensions of the Pinlite (Pinlite, Inc.) seven-segment display tube are shown in Fig. 4-25B. This is also an incandescent device with the following ratings:

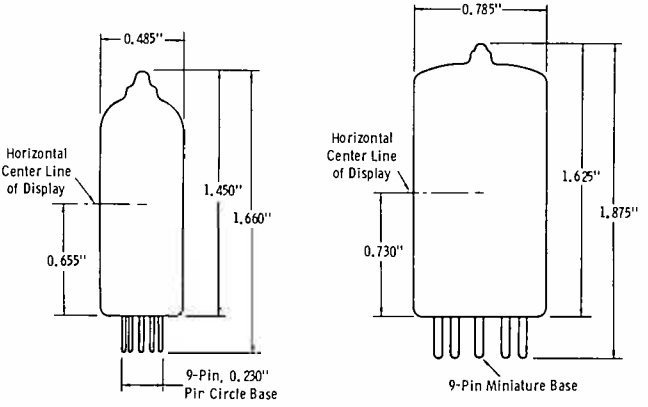
O Type: 1.5 to 3 volts at 8 mA per segment
R Type: 2 to 3 volts at 4.3 mA per segment

The display is viewed from the top. Note the different configuration for segment lettering as contrasted with the more conventional type of Fig. 4-23A.

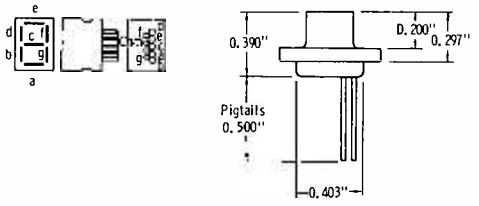
The Tung-Sol Digivac Types DT1704B (Fig. 4-25C) and DT1705D, the Nippon Electric (NEC) Type DG12E/LD915, and the Sylvania Type 8894 are all low-voltage, low-power vacuum fluorescent seven-segment readouts. In these displays, the filament requirement is constant at 1.6 volts and 45 milliamperes. The individual segments require 12 to 25 volts at 100 to 300 microamperes.

The LED Readout

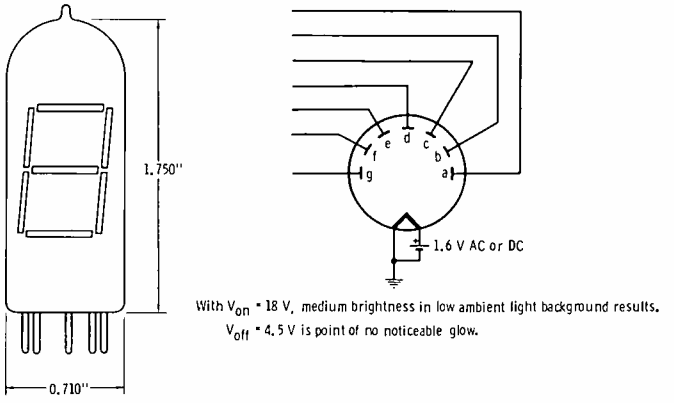
The *light-emitting diode* (*LED*) is an extremely popular display device, due to the very low voltage requirement, ruggedness, and long life. The LED is a semiconductor device made either of gallium arsenide (GaAs), gallium arsenide phosphide (GaAsP), or gallium phosphide (GaP). The GaAs diode produces *infrared* light at around 9000 angstroms when forward-biased. It is used with photodetectors for such applications as card readers, etc., where the human eye is not used. The GaAsP diode produces visible light



(A) RCA Numitron.



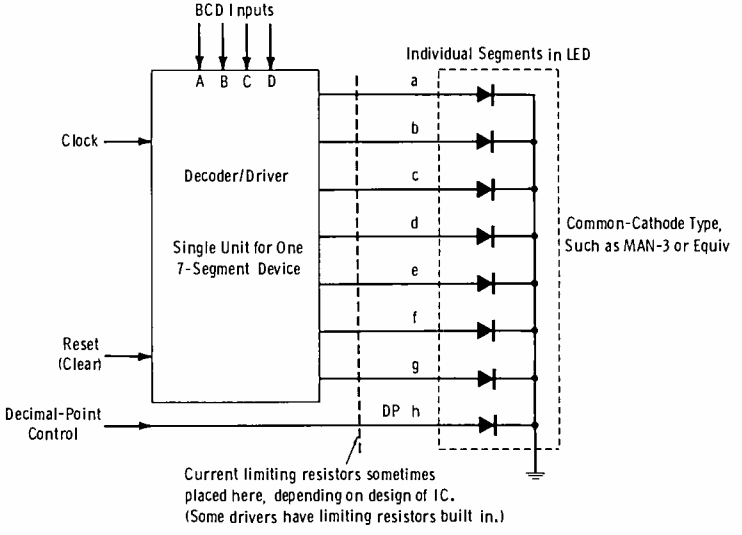
(B) Pinlite, Inc. Series O and R.



(C) Tung-Sol Digivac Type DT1704B.

Fig. 4-25. Typical seven-segment display tubes.

(A) Display in hand-held calculator.



(B) Decoder/driver and display.

Fig. 4-26. Seven-segment LEDs.

(red) in the 6600-angstrom range. The GaP diode can be processed to produce other colors, such as green or yellow.

The direct-reading LED is a seven-segment device (Fig. 4-23A). Although its numbers are smaller than in some of the gaseous counterparts (from 0.1 to 0.6 inch in height), they are generally clear, sharp, and easy to read.

Fig. 4-26A illustrates the appearance of an LED display in a calculator unit. Although each diode has segmented characters, the brightness (and smallness) results in a display that appears to be continuous. The only way you can observe the segments is by dimming the display and observing it through a magnifying glass.

Fig. 4-26B shows a typical configuration of a decoder-driver and LED display. The voltage is 1.5 to 5 volts, and the current required is from 10 to 30 mA per segment, depending on the type of LED. The unit is normally on a 14-pin base similar to a 14-pin IC. Some units have the decoder-driver included in the same package. Also, units are available that incorporate multiple-digit displays with four or more digits.

The Monsanto MAN-1 is a seven-segment LED display device on a 14-pin IC base. Its electrical characteristics and physical dimensions are shown in Fig. 4-27A. Note that this is a common-anode device, as contrasted with the drawing of Fig. 4-26B. The numbers are almost 0.3 inch high, with a 10° slant configuration.

Fig. 4-27B illustrates the MAN-3, a smaller display (character height about 0.1 inch) than the MAN-1. The forward diode current is 5 mA (per segment) at 1.7 to 2 volts. Both the MAN-1 and MAN-3 types are GaAsP diodes. The MAN-3 has a common cathode.

The physical characteristics of a larger LED are shown in Fig. 4-27C. The characters may be either 0.3 or 0.6 inch high, and they are available in red, green, or yellow color. This is a GaP diode, with a typical voltage of 2 volts at 10 mA per segment. Maximum current for the number 8 (seven segments) is thus $7 \times 10 = 70$ mA. These units are quite flexible in application, being available in common-anode, left-decimal or common-cathode, right-decimal versions. (Review Fig. 4-23A.)

Blanking of Nonsignificant Zeros

Fig. 4-28 shows the logic diagram of a typical decade counter/divider with decoded seven-segment display outputs. All of this circuitry is contained in a single 16-pin integrated-circuit chip. Note that in addition to the lamp test input, a ripple blanking input (RBI) and ripple blanking output (RBO) are provided.

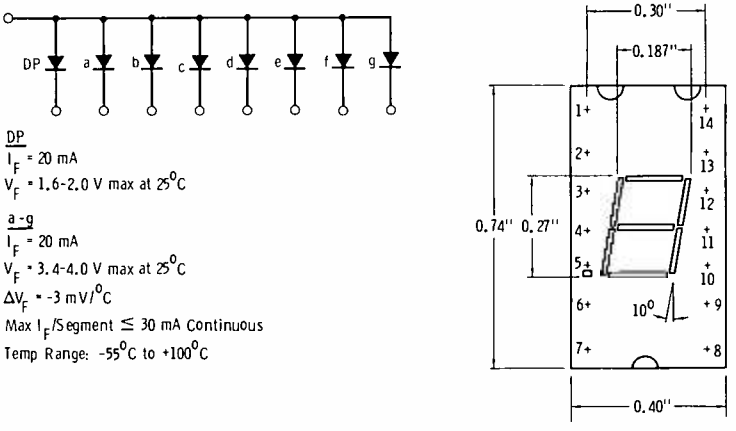
This type of circuitry is normally used with a display system that has a fixed decimal point (Fig. 4-24).

The inputs to the CD4033A IC are CLOCK, RESET, CLOCK ENABLE, RBI (ripple blanking input), and LAMP TEST as shown in Fig. 4-28. The outputs are C_{out} (carry out), RBO (ripple blanking output), and the seven decoded outputs (a, b, c, d, e, f, g). A high reset signal clears the decade counter to its zero count. The counter is advanced one count at the positive clock-signal transition if the clock-enable signal is low. Counter advancement by way of the clock line is inhibited when the clock-enable signal is high. A timing diagram for the CD4033A is shown in Fig. 4-29.

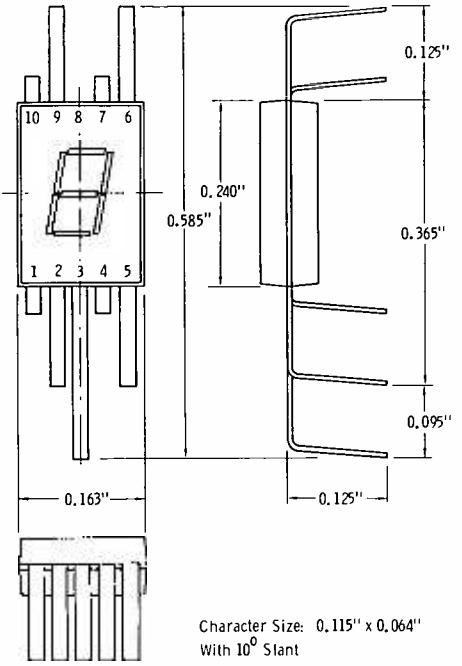
The carry-out (C_{OUT}) signal completes one cycle for every ten clock input cycles and is used to clock the succeeding decade directly in a multidecade counting chain. The seven decoded outputs (a, b, c, d, e, f, g) illuminate the proper segments in a seven-segment display device used to present the decimal digits 0 to 9. The seven-segment outputs go high on selection.

This unit has provisions for automatic blanking of the nonsignificant zeros in a multidigit decimal number. This feature results in an easily readable display consistent with normal writing practice. For example, the number 0050.0700 in an eight-digit display would

be displayed as 50.07. Zero suppression on the integer side is obtained as follows: The RBI terminal of the CD4033A associated with the most significant position in the display is connected to a low-level voltage; the RBO terminal of the same stage is connected to the



(A) Schematic and dimensions of MAN-1.



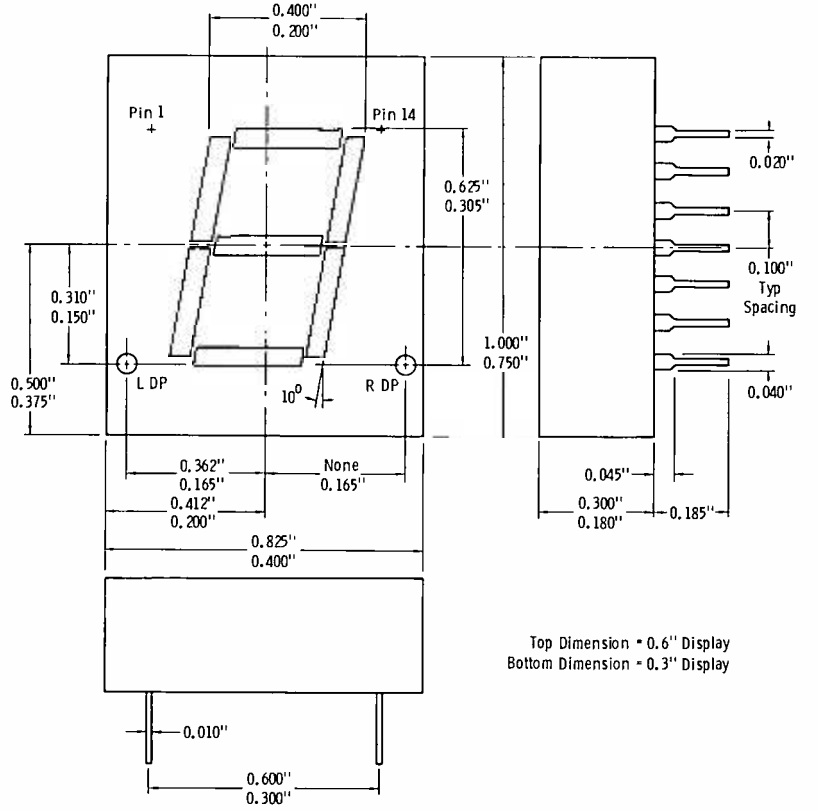
(B) Physical dimensions of MAN-3.

Fig. 4-27. Typical

RBI terminal of the CD4033A in the next lower significant position in the display. This procedure is continued for each succeeding CD4033A of the integer side of the display.

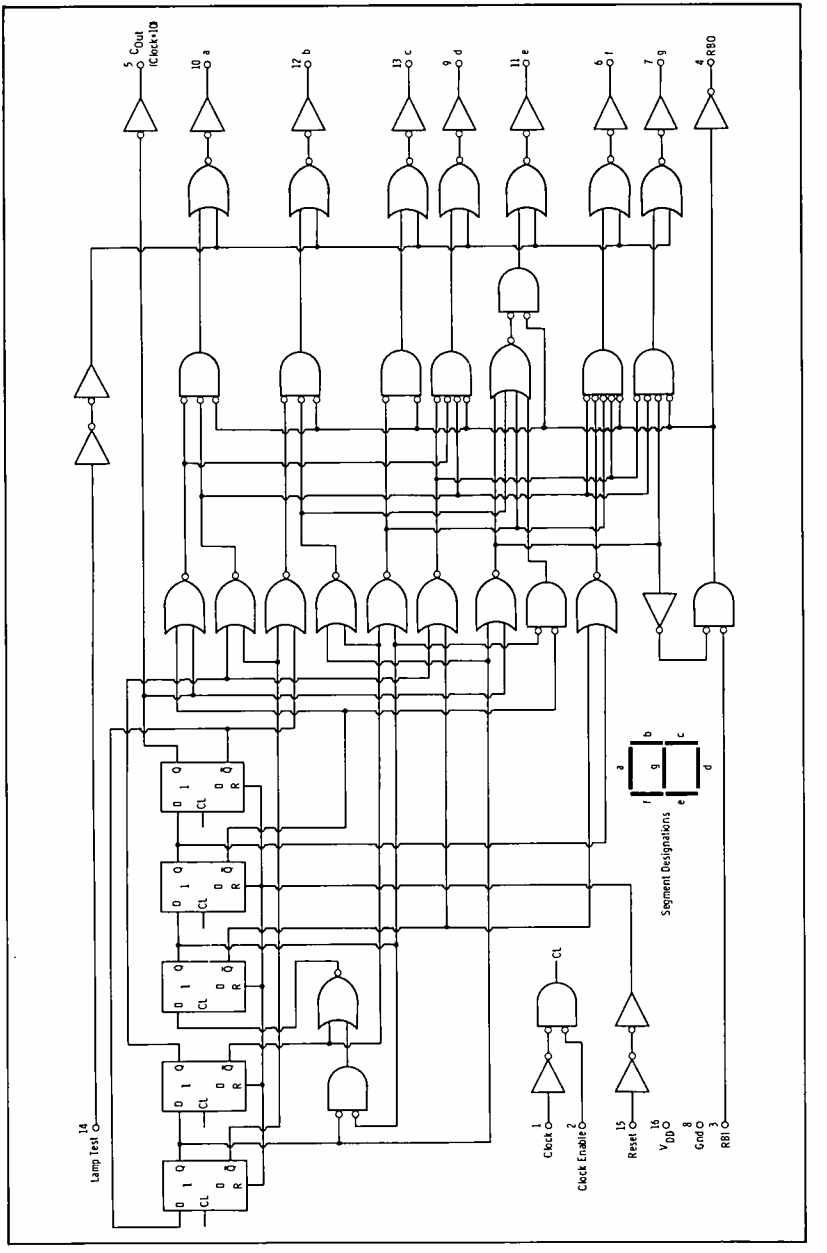
On the fraction side of the display, the RBI terminal of the CD4033A associated with the least significant position is connected to a low-level voltage, and the RBO terminal of the same CD4033A is connected to the RBI terminal of the CD4033A in the next more significant position. This procedure is continued for all CD4033As on the fraction side of the display.

In a purely fractional number (e.g., 0.7346), the zero immediately preceding the decimal point can be displayed by the connection of the RBI terminal of that stage to a high-level voltage (instead of to the RBO terminal of the next more significant stage). Similarly, the zero in a number such as 763.0 can be displayed by the connection of the RBI terminal of the CD4033A associated with it



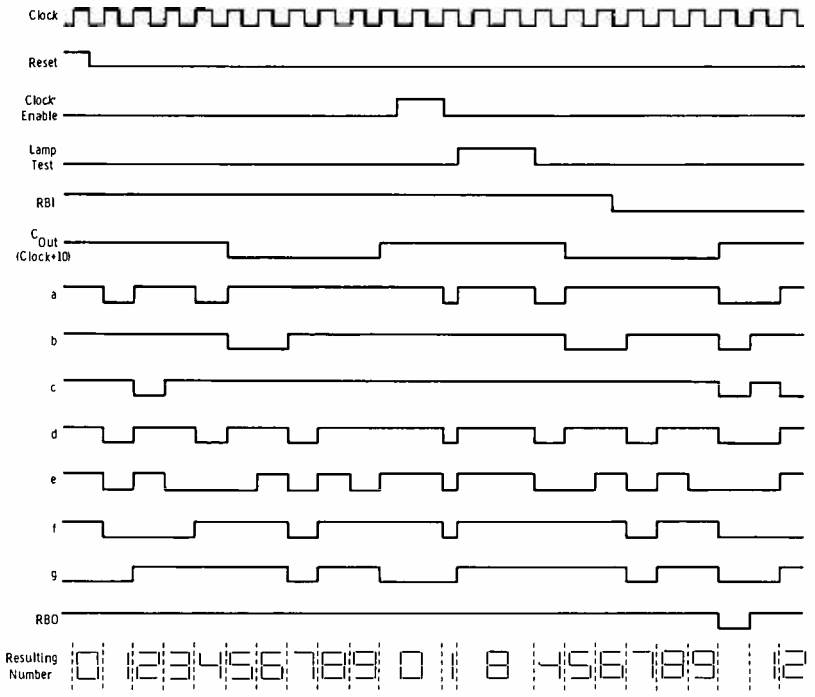
(C) Solid-State Systems, Inc. LED display.

LED display devices.



Courtesy RCA

Fig. 4-28. Logic diagram for CD4033A decade counter/divider with decoded seven-segment display outputs.



Courtesy RCA

Fig. 4-29. Timing diagram for CD4033A counter.

to a high-level voltage. Ripple blanking of nonsignificant zeros provides an appreciable savings of display power.

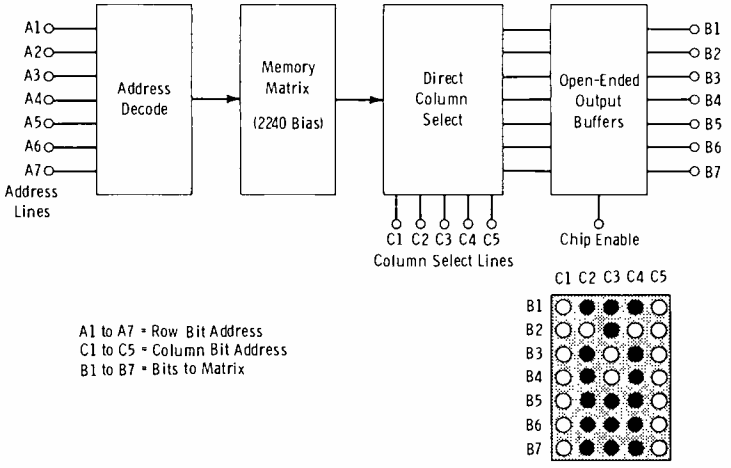
The CD4033A has a lamp-test input that, when connected to a high-level voltage, overrides normal decoder operation and enables a check to be made of possible display malfunctions by putting the seven outputs in the high state. Note that this is just the opposite of the IC shown in Fig. 4-23C, which requires grounding the L/T input.

Matrix Displays With LEDs

Plain (no numbers) LEDs are sometimes used in a matrix to form larger displays for viewing at a greater distance. Numbers and/or letters are formed by lighting the specified LEDs for each particular character. The system normally employs a *strobing* technique, which means that a memory element is "sensed" by a pulse to determine what information, if any, exists, and to prepare the element for readout.

From the foregoing, it can be seen that a matrix is sampled at a high rate and is pulsed in action. The LED has an extremely fast

(*A*) *Appearance of display.*



(*B*) *Block diagram of display.*

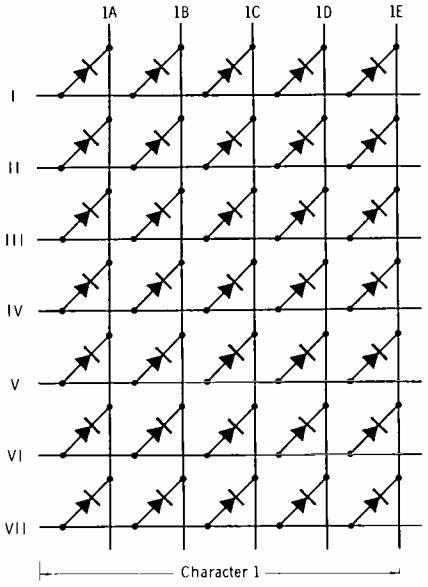
Fig. 4-30. Principle of LED matrix for alphanumeric display.

pulse response and is well suited for this application. It should be pointed out that some seven-segment displays are strobed in the same fashion. The principle is the same as that described below.

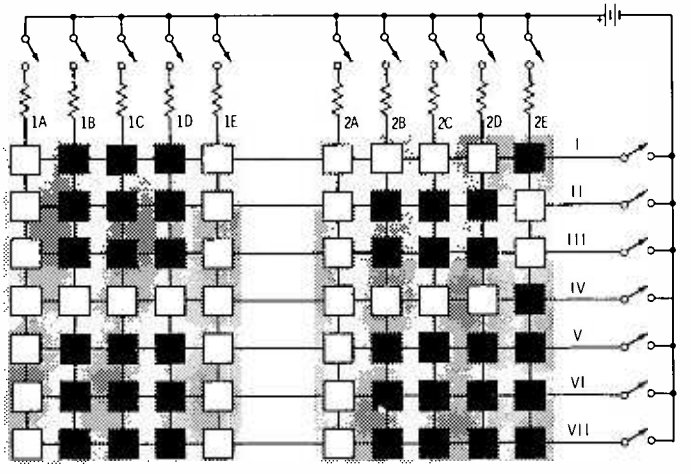
Fig. 4-30A shows the appearance of a typical 5×7 LED matrix. This means five dots are available horizontally and seven dots are available vertically. Each "dot" is an individual LED. Fig. 4-30B is a block diagram illustrating the basic idea for the matrix display. A "row" and "column" address such as that provided by the ASCII code (Chapter 5) is used. Since each basic character of the alphanumeric display is (typically) a 5×7 matrix of LEDs, the LEDs are electrically accessed in row and column fashion.

Fig. 4-31A illustrates the LED hookup for a single character in a Hewlett-Packard alphanumeric display. In Figs. 4-31A and 4-31B, row address is signified by Roman numerals, and column address is signified by letters A through E. There are five connections for the five columns associated with the LED cathodes, and seven connections for the seven rows associated with the LED anodes.

The resulting X-Y accessing, combined with time sharing, provides control of all points in the matrix. In the display of Fig. 4-31,



(A) Electrical arrangement of single character.



(B) Arrangement for vertical strobing.

Fig. 4-31. Alphanumeric display using LED matrix.

the arrays are scanned from top to bottom, row by row, with the encoded signal for each row of all characters appearing simultaneously. Thus, the characters are formed by time-sequentially selecting rows and activating the appropriate diodes in each row. This is vertical strobing, as indicated for the two-character display of Fig.

4-31B. The strobe rate is normally above 100 Hz to avoid flicker. In practice, rates may be between 250 Hz and 10 kHz.

EXERCISES

Q4-1. How can a JK flip-flop be used as a toggle?

Q4-2. What is the advantage of a JK flip-flop over an RS flip-flop?

Q4-3. How can a flip-flop be called a "memory"?

Q4-4. Contrast the pulse-triggered master-slave flip-flop with the edge-triggered master-slave flip-flop.

Q4-5. Does a binary divider always divide by an even number?

Q4-6. What is a shift register?

Q4-7. What is a major difference between a decoder/driver for gaseous "sheet" (Nixie-type) display tubes and one used to drive segmented display devices?

Q4-8. What is a "floating decimal"?

CHAPTER 5

Basic Codes

A code is a system that represents a set of symbols by means of another (different) set of symbols. You have studied the straight 8-4-2-1 binary code, the binary-coded decimal (bcd) code, and the octal code in Chapter 2. Fig. 5-1 shows the expression of decimal 123 in the different codes covered thus far.

Any binary code is made up of two symbols, 0 and 1. But a code must be separated in your thinking from a *number system* based on the radix 2. A code is not necessarily *weighted*, as it must be in a number system. We will distinguish codes in general classifications as follows:

1. Weighted
2. Nonweighted
3. With parity bits for error detection
4. Error-correcting

It is not necessary to memorize codes, but it is important to understand the principles of coding. Different systems use different codes, and it would be impractical and unnecessary to list all codes used today.

5-1. WEIGHTED CODES

The straight binary code is weighted by powers of 2. Thus, 8-4-2-1 reveals that if more bits are involved, you know the next bit to the left is 16, the next 32, etc.

Similarly, a 7-4-2-1 code may be used. Note here that you have a simple weighting series that begins with 1 and increases regularly as: plus one unit, plus two units, plus three units, plus four units, etc. Thus, the decimal equivalent of 1111 for the 7-4-2-1 code is $7 + 4 + 2 + 1 = 14$. For five digits, you know that the next digit to

System	Code
Decimal	123
Binary (Radix 2)	1111011
Binary-Coded Decimal (BCD)	0001 0010 0011
Binary-Coded Octal (BCO)	001 111 011
Octal (Radix 8)	173

Fig. 5-1. Expression of decimal 123 in different codes.

the left would represent eleven. For this code 11111 would be $11 + 7 + 4 + 2 + 1 = 25$.

The coding discussed above is termed *regularly* weighted because a rule can be stated for the formation of the weighting values. *Arbitrarily* weighted codes also employ weighting values, but no rule can be evolved that describes their formation. For example, the 4-2-2-1 code simply means that 1111 would be $4 + 2 + 2 + 1 = 9$. Such arbitrarily weighted codes are often used. It is immaterial

Table 5-1. Decimal Codes

COMMON WEIGHTED DECIMAL CODES					
	8421	6311	5211	3321	51111
0	0000	0000	0000	0000	00000
1	0001	0001	0001	0001	00001
2	0010	0011	0100	0010	00011
3	0011	0100	0110	0011	00111
4	0100	0101	0111	0101	01111
5	0101	0111	1000	1010	10000
6	0110	1000	1001	1100	11000
7	0111	1001	1011	1101	11100
8	1000	1011	1110	1110	11110
9	1001	1100	1111	1111	11111

NONWEIGHTED DECIMAL CODES		
	"Excess Three"	"Two-Out-of-Five"
0	0011	00011
1	0100	00101
2	0101	00110
3	0110	01001
4	0111	01010
5	1000	01100
6	1001	10001
7	1010	10010
8	1011	10100
9	1100	11000

what a fifth-digit weight might be, since it is never used unless spelled out by the code designer.

Table 5-1 lists other common weighted codes, as well as two of the most common nonweighted codes.

5-2. NONWEIGHTED CODES

A nonweighted code has no weighting value. Each group of bits, or binary word, represents some given quantity.

Note the *excess-3* code in Table 5-1. In pure binary, 0 is represented by 0000. The excess-3 code simply adds 3 to obtain 0011 to represent decimal zero. This eliminates the problem of a string of zeros that contain "no information" (discussed in Section 5-3 on error detection). All following values in this code are the pure binary code plus 3.

This code is widely used for several reasons. To see why, get some scratch paper and play around with this code as follows: Take any word in the table, such as 1010 (decimal 7). Invert each of its bits (take the ones complement). Now you have 0101 (decimal 2). Note that $7 + 2 = 9$. You will find that the sum of any excess-3 word and its complement (in decimal values) will be nine. This property is termed *self-complementary*. It is widely used in calculators to obtain subtraction by adding. The rules for excess-3 addition are not the same as for straight binary addition. These rules will not be discussed in this text, since they do not apply to communications circuitry.

The "two-out-of-five" code is discussed in the following section.

5-3. ERROR DETECTION

The binary system has no "natural" symptoms of trouble, such as distortion, to warn the operator of problems. What occurs is a wrong decision or wrong operation of controlled devices. It is important to provide some means of detecting a possible error.

One of the simplest methods of error detection that is often used in broadcast systems is the *inverted,* or *complementary,* technique. For example, if a code word is 0111001, it may be followed with the inverted (complementary) code 1000110. If the complement does not check, the decision or operation is not made, and the entire procedure is repeated. If several such repetitions do not result in a decision, the circuitry may shut down, or an error indicator may light.

Observe the *two-out-of-five* code in Table 5-1. Note that the code is arranged so that the number of 1's representing a specific number is always two. Thus, if any 0 should be erroneously changed to a 1 (such as by a noise impulse), or any 1 should be missed, an error is

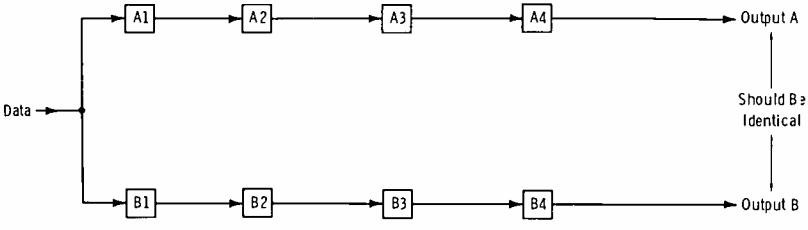
automatically indicated. This is just one type of the many two-out-of-five codes possible.

A given data-processing operation can take the form of redundant circuitry as shown in Fig. 5-2A. Since each path is performing the same function, the outputs should be identical. If a discrepancy occurs, an error is indicated.

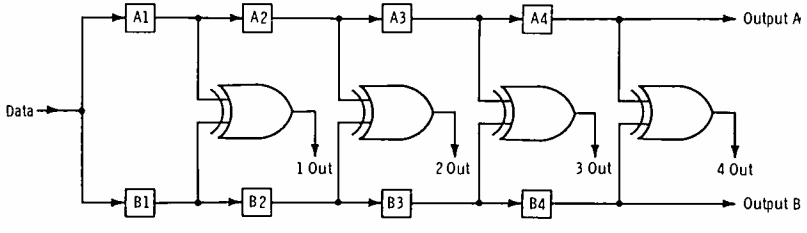
The exact stage in which the error occurs can be revealed by the arrangement of Fig. 5-2B. Recall that an Exclusive OR gate has an output only when both inputs are of opposite value. (In this application, the Exclusive OR gates are called *comparators.*) If both functions are identical, as they should be, no error output occurs. Obviously, parallel circuitry of this kind doubles the hardware cost of any particular installation.

A *parity bit* is an added (redundant) bit that contains no message information but is used strictly for error detection. If you take the straight binary 8-4-2-1 code and add another bit for parity, you have an 8-4-2-1-0 code. The added "0" column contains no message information but is an error-detecting column.

The added parity column can be arranged to make the total number of 1's in a binary word an odd or even count. See Table 5-2 and note that for an odd parity, the binary equivalent of decimal 0, or 0000, becomes 00001. This has an advantage over even parity where decimal zero becomes 00000, which is normally to be avoided. A string of 0's always leads to the question of whether the "lack of information" is a pause between pulses, or missed "1" bits.



(A) Parallel operation for accuracy check.



(B) Exclusive OR checks for each step.

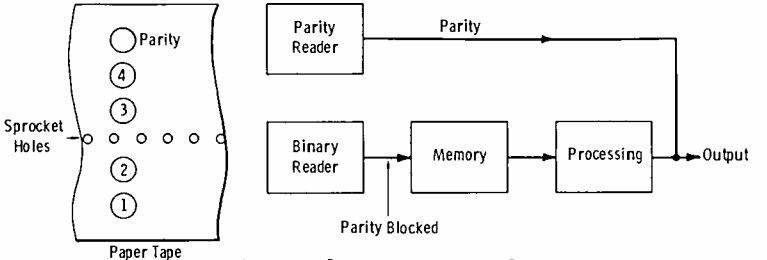
Fig. 5-2. Parallel-operation methods of error detection.

Table 5-2. Odd and Even Parity Check

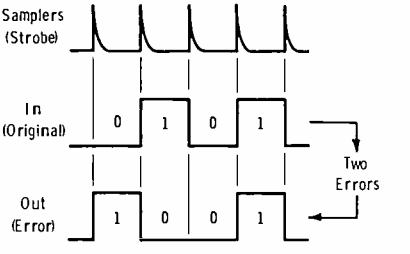
Decimal	Odd Parity 8	4	2	1	P ↓ 0	Even Parity 8	4	2	1	P ↓ 0
0	0	0	0	0	1	0	0	0	0	0
1	0	0	0	1	0	0	0	0	1	1
2	0	0	1	0	0	0	0	1	0	1
3	0	0	1	1	1	0	0	1	1	0
4	0	1	0	0	0	0	1	0	0	1
5	0	1	0	1	1	0	1	0	1	0
6	0	1	1	0	1	0	1	1	0	0
7	0	1	1	1	0	0	1	1	1	1
8	1	0	0	0	0	1	0	0	0	1
9	1	0	0	1	1	1	0	0	1	0

For odd parity, the parity bit is such that the total 1's (including parity) result in an odd number.
For even parity, the parity bit is such that the total 1's (including parity) result in an even number.

The parity check is not similar to the two-out-of-five code discussed previously, since in that code, two 1's always occur. The parity bit follows a separate routing as shown by Fig. 5-3A. This example shows a 4-bit paper tape with an added parity function. Bits 1 through 4 are arranged vertically; a 1 is represented by a



(A) Typical routing of parity bits.



(B) Double error not detected.

Fig. 5-3. Parity check.

punched hole, a 0 by no punched hole (blank). These are "read out" by the tape reader and passed to the digital system. The parity row is blocked from this path, and is read by a separate parity reader and reinserted at the end junction. In this way, the parity bit is not subject to system errors that may occur in the message bits.

Note from Fig. 5-3B that a parity check of this nature will not detect a double error in a binary word. In this example, there are still two 1's in the message byte, and the parity (odd or even) will indicate that the message is correct. Obviously, the chances of having two errors are much less than the chances of having one error.

5-4. ERROR CORRECTION

Review Fig. 5-2B. The outputs of the comparators (Exclusive OR gates) may be fed back to the appropriate circuit and used to correct the error automatically. The action is a simple inversion, since if the bit is not a 0 it should be a 1, and vice versa. Obviously, if the flip-flop or other circuit is malfunctioning, the error is not corrected, but the faulty component is indicated. Parallel operation is the most reliable form of error detection and correction, but it is also the most costly.

A word-by-word error-correction code that may be used is based on the parity-check principle, but it is slightly more sophisticated than that described in Section 5-3. This simply means that the hardware is more complex and hence more costly, but the added expense is not as great as for parallel operation.

Table 5-3 shows an example of such a code for the 8-4-2-1 binary word. Three parity bits (P1, P2, and P3) are used as redundant information, with the following relationship:

P1 checks 4-2-1 bit positions
P2 checks 8-2-1 bit positions
P3 checks 8-4-1 bit positions

An even parity check is used. This means that P1 compares the number of bits in the 4-2-1 columns, and the value of P1 is such that the total number of 1's, including P1, is an even number. The value of P2 is such that an even number of 1's occurs in the 8-2-1-P2 columns. The value of P3 is such that an even number of 1's occurs in the 8-4-1-P3 columns.

Fig. 5-4 shows the practical application of locating the error in a given binary word (first four digits) and correcting that error. Study this procedure until you understand it, and you will be able to use any such code that you might encounter in practice.

In the example above, a P1 failure means that either the 4, 2, or 1 bit is in error. The P2 failure means that either the 8, 2, or 1 bit

Table 5-3. 7-Bit Parity Code. Word-by-Word Error Correction for 4-Bit Message (Even Parity)

	8	4	2	1	P1 (4-2-1 Weight)	P2 (8-2-1 Weight)	P3 (8-4-1 Weight)
0	0	0	0	0	0	0	0
1	0	0	0	1	1	1	1
2	0	0	1	0	1	1	0
3	0	0	1	1	0	0	1
4	0	1	0	0	1	0	1
5	0	1	0	1	0	1	0
6	0	1	1	0	0	1	1
7	0	1	1	1	1	0	0
8	1	0	0	0	0	1	1
9	1	0	0	1	1	0	0

is in error. But the P3 check does not fail, which indicates that the 8, 4, and 1 bits have no error. This leaves only the 2 bit that could be in error.

In practice, the following applies:

1. If all three checks fail, the error is in the 1 bit.
2. If P2 and P3 fail, the error is in the 8 bit.

1011100* contains an error as proven below:

8	4	2	1	P1	P2	P3	Even Parity
	0	1	1	1			P1 check fails because total number of ones is odd.
1		1	1		0		P2 check fails because total number of ones is odd.
1	0		1			0	P3 check ok because total number of ones is even.
							1011100 as decoded is decimal 11. The error is corrected by inverting the 2 bit, forming 1001100 (decimal 9, correct).

Check:

1	0	0	1	1	0	0	
	0	0	1	1			P1 checks ok.
1		0	1		0		P2 checks ok.
1	0		1			0	P3 checks ok.

*The binary word 1011100 is decimal 11. (The last three bits are parity checks.)

Fig. 5-4. Example of finding the error in 1011100.

3. If P1 and P2 fail, the error is in the 2 bit. (See Fig. 5-4.)
4. If P1 and P3 fail, the error is in the 4 bit.
5. If only one check fails (either P1, P2, or P3), then the error is in the parity bit itself.

You can observe Rule 5 by using the chart of Table 5-3 and introducing a single parity-bit error. Then follow the procedure of Fig. 5-4. This provides good practice.

As with other parity-check schemes covered so far, the code shown in Table 5-3 is useless (invalid) if *two* errors occur. Another type of code (sometimes termed "two-dimensional") adds another row of parity bits at the bottom of the columns so that two errors may be detected.

Visualize this scheme as follows: Call the parity checks of Table 5-3 the "horizontal" (word) check bits. Add another parity row at the bottom of Table 5-3, one parity bit under each column, for the 8-4-2-1 positions (Table 5-4). Call these the "vertical" (column) check bits. In this example, even parity would be used. Therefore, the vertical parity bits would be 0001, as shown in the bottom row of Table 5-4.

Assume that an error existed in decimal 5 (0101) such that the 8 and 4 positions were interchanged (1001). Even though the horizontal check for P3 shows no error (double error does not change the total number of 1's), the vertical check will show errors in the 8 and 4 positions. The limitation here is that only the error in the columns is indicated; the word in which the error occurred is not indicated. Such tradeoffs must be considered by the digital designer in choosing the type of operation best suited to a given application. This particular application would be for error detection only. Error correction would not be possible.

Table 5-4. Two-Dimensional Parity

	8	4	2	1	P1	P2	P3
0	0	0	0	0	0	0	0
1	0	0	0	1	1	1	1
2	0	0	1	0	1	1	0
3	0	0	1	1	0	0	1
4	0	1	0	0	1	0	1
5	0	1	0	1	0	1	0
6	0	1	1	0	0	1	1
7	0	1	1	1	1	0	0
8	1	0	0	0	0	1	1
9	1	0	0	1	1	0	0
Vert Parity	0	0	0	1			

Note carefully that in this example of weights 8 and 4 being interchanged, P1 and P2 would fail, with P3 showing no error. This normally indicates that weight 2 contains the error. However, you know that a double error has occurred (weights 8 and 4), thus rendering the code invalid for double error. The only error *detection* is from the added vertical-parity row.

5-5. THE ASCII CODE

The digital system must be able to work with alphabetic characters as well as numeric characters; this combination is termed an *alphanumeric* code. Attempts have been made to create a standard, universal code. The ASCII (American Standard Code for Information Interchange) is widely used in communications and broadcast services. This code contains seven bits (with an extra eighth bit for parity if desired), with 128 possible combinations ($2^7 = 128$). The seven bits allow the expression of all numerical values, plus the alphabet, punctuation marks, symbols, and special telephone and teletypewriter abbreviations.

Fig. 5-5 shows the entire code. Note that the high-order bits (b7, b6, b5) increase in value by straight binary addition. Likewise, the low-order bits (b4, b3, b2, b1) increase in value by straight binary addition. Thus, decimal numbers 0 through 9 are represented by column 011 (column 3) and rows 0000 through 1001 (decimal 0–9). Therefore, the ASCII code is compatible with systems designed to use the 8-4-2-1 bcd code.

Any single character (or function) can be described by its particular combination of high-order (column) and low-order (row) bits. For example, find the code for the capital letter Y as illustrated by Fig. 5-6.

Because of the grouping of control and character functions, the circuitry must analyze only the first two bits of a code to determine whether it is a control (both zeros) or a character (not both zeros) function. Note from Fig. 5-5 that columns 000 (column 0) and 001 (column 1) are strictly control functions. This code is widely used in card or paper-tape punching devices, readout printers, and video-monitor display systems (such as the one in Fig. 1-6, Chapter 1).

ASCII Control Character Definitions

The definitions that follow are recommended by Sectional Committee X3 of the American National Standards Institute (ANSI); however, several versions are in existence. (The terms are listed in the order in which they appear in Fig. 5-5.)

NUL (Null)—The all-zeros character which may serve to accomplish time fill and media fill.

BITS				b_7 → b_6 → b_5 →	0_0	0_1	0_1	0_1	1_0	1_0	1_1	1_1
b_4 ↓	b_3 ↓	b_2 ↓	b_1 ↓	COL / ROW	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	¦
1	1	0	1	13	CR	GS	-	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	_	o	DEL

Fig. 5-5. ASCII encoding chart.

SOH (Start of Heading)—A communication control character used at the beginning of a sequence of characters that constitute a machine-sensible address or routing information. Such a sequence is referred to as the heading. An STX character has the effect of terminating a heading.

STX (Start of Text)—A communication control character that precedes a sequence of characters that is to be treated as an entity and entirely transmitted through to the ultimate destination. Such a

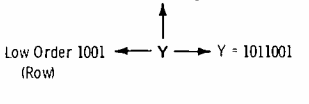


Fig. 5-6. Capital letter Y in ASCII code.

sequence is referred to as *text*. STX may be used to terminate a sequence of characters started by SOH.

ETX (*End of Text*)—A communication control character used to terminate a sequence of characters started with STX and transmitted as an entity.

EOT (*End of Transmission*)—A communication control character used to indicate the conclusion of a transmission, which may have contained one or more texts and any associated headings.

ENQ (*Enquiry*)—A communication control character used in data communication systems as a request for a response from a remote station. It may be used as a "Who Are You" (WRU) to obtain identification or to obtain station status, or both.

ACK (*Acknowledge*)—A communication control character transmitted by a receiver as an affirmative response to a sender.

BEL (*Bell*)—A character used to indicate that human attention is needed. It may control alarm or attention devices.

BS (*Backspace*)—A format effector that controls the movement of the printing position one printing space backward on the same printing line. (Applicable also to display devices.)

HT (*Horizontal Tabulation*)—A format effector that controls the movement of the printing position to the next in a series of predetermined positions along the printing line. (Applicable also to display devices and the skip function on punched cards.)

LF (*Line Feed*)—A format effector that controls the movement of the printing position to the next printing line. (Applicable also to display devices.)

VT (*Vertical Tabulation*)—A format effector that controls the movement of the printing position to the next in a series of predetermined printing lines. (Applicable also to display devices.)

FF (*Form Feed*)—A format effector that controls the movement of the printing position to the first predetermined printing line on the next form or page. (Applicable also to display devices.)

CR (*Carriage Return*)—A format effector that controls the movement of the printing position to the first printing position on the same printing line. (Applicable also to display devices.)

SO (*Shift Out*)—A control character indicating that the code combinations that follow shall be interpreted as outside of the character set of the standard code table until a *Shift In* character is reached.

SI (*Shift In*)—A control character indicating that the code combinations that follow shall be interpreted according to the standard code table.

DLE (*Data Link Escape*)—A communication control character that will change the meaning of a limited number of contiguously following characters. It is used exclusively to provide supplementary controls in data communication networks.

DC1, DC2, DC3, DC4 (Device Controls)—Characters for the control of ancillary devices associated with data processing or telecommunication systems, more especially for switching devices "on" or "off." (If a single "stop" control is required to interrupt or turn off ancillary devices, DC4 is the preferred assignment.)

NAK (Negative Acknowledge)—A communication control character transmitted by a receiver as a negative response to the sender.

SYN (Synchronous Idle)—A communication control character used by a synchronous transmission system in the absence of any other character to provide a signal from which synchronism may be achieved or retained.

ETB (End of Transmission Block)—A communication control character used to indicate the end of a block of data for communication purposes. ETB is used for blocking data where the block structure is not necessarily related to the processing format.

CAN (Cancel)—A control character used to indicate that the data with which it is sent are in error or are to be disregarded.

EM (End of Medium)—A control character associated with the sent data which may be used to identify the physical end of the medium or the end of the used, or wanted, portion of information recorded on a medium. (The position of this character does not necessarily correspond to the physical end of the medium.)

SUB (Substitute)—A character that may be substituted for a character which is determined to be invalid or in error.

ESC (Escape)—A control character intended to provide code extension (supplementary characters) in general information interchange. The escape character itself is a prefix affecting the interpretation of a limited number of contiguously following characters.

FS (File Separator), GS (Group Separator), RS (Record Separator), and US (Unit Separator)—These information separators may be used within data in optional fashion, except that their hierarchical relationship shall be: FS is the most inclusive, then GS, then RS; and US is least inclusive. (The content and length of a file, group, record, or unit are not specified.)

SP (Space)—A normally nonprinting graphic character used to separate words. It is also a format effector that controls the movement of the printing position, one printing position forward. (Applicable also to display devices.)

DEL (Delete)—This character is used primarily to erase or obliterate erroneous or unwanted characters in perforated tape. (In the strict sense, DEL is not a control character.)

Modified ASCII for Television Use

The video character generator is a real workhorse in modern television applications. Alphanumeric characters may be made to crawl

(usually from right to left across the bottom fourth of the picture) to present statistics (such as a baseball player's batting average), election returns, news or weather bulletins, etc. Or, the display may be made to roll (usually from bottom to top, for such information as show credits, etc.

Fig. 5-7 illustrates a typical decoding chart for such systems, showing a modification of the standard ASCII chart. Note that only capital letters are used; columns 6 and 7 are deleted entirely. In column 1, row 3, CRSR abbreviates the word 'cursor." The cursor indicates the position of the next character to be typed on the keyboard and appears as a white square on the monitor. It may be turned off if desired, as indicated in Table 5-5, which shows typical control functions and their definitions. Note that the cursor control replaces

BITS b_7 b_6 b_5 →				0_0	0_1	1_0	0_1	1_0	1_0	1_1	1_1
b_4	b_3	b_2	b_1	COL / ROW	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL		SP	0	`	P		
0	0	0	1	1	SOH	FLASH	!	1	A	Q		
0	0	1	0	2		OPEN	"	2	B	R		
0	0	1	1	3	ETX	CRSR	#	3	C	S		
0	1	0	0	4	EOT	CLOSE	$	4	D	T		
0	1	0	1	5			%	5	E	U		
0	1	1	0	6			&	6	F	V		
0	1	1	1	7	BEL		'	7	G	W		
1	0	0	0	8	CRSR LEFT		(	8	H	X		
1	0	0	1	9	CRSR RIGHT	EM	)	9	I	Y		
1	0	1	0	10	LF	ERASE	*	:	J	Z		
1	0	1	1	11	CRSR DOWN		+	;	K	[		
1	1	0	0	12	CRSR UP	CRSR HOME	,	<	L	~		
1	1	0	1	13	CR	CRSR NEW LINE	-	=	M	]		
1	1	1	0	14		SND	.	>	N	^		
1	1	1	1	15		SEND MSG	/	?	O	_		

Courtesy A. B. Dick Co.

Fig. 5-7. Typical ASCII decoding chart for television broadcast use.

Table 5-5. Definitions of Control Functions*

Special Character	Function
Flash	Causes subsequent characters to flash.
Open	Moves cursor line down one increment and displaces bottom line (for systems so equipped).
Close	Deletes line above cursor and moves remainder of display up one increment (for systems so equipped).
ETX	End of transmission for up/down shift, external data sink encoding. Does not enter memory.
SND MSG	Outputs complete memory content to external data sink.
SND LINE	Outputs cursor line only to external data sink.
Erase	Deletes display and memory content.
RM	Stop code. Enters memory.
Home	Moves cursor to home position.
CRSR ↑ ← → ↓	ON-OFF switch for cursor. Moves cursor up one line. Moves cursor left one increment. Moves cursor right one increment. Moves cursor down one line.
Crawl/Nor/Roll	Roll/crawl mode selector (if so equipped).
Fast/Med/Slow (Roll)	Speed select for roll mode (if so equipped).
Fast/Med/Slow (Crawl)	Speed select for crawl mode (if so equipped).
Load Req	Accesses external data sink.

* A. B. Dick Videograph Series 990

BS, HT, VT, FF, FS, and GS in the standard ASCII code. "Home" position is the upper left of the "page."

The functions of crawl and roll are selected on a separate group of control buttons at the keyboard unit.

Video character generator systems are more fully covered in Chapter 9.

5-6. THE SMPTE CODE

The editing of video tape and the synchronization of audio tape have been greatly improved by the replacement of former methods

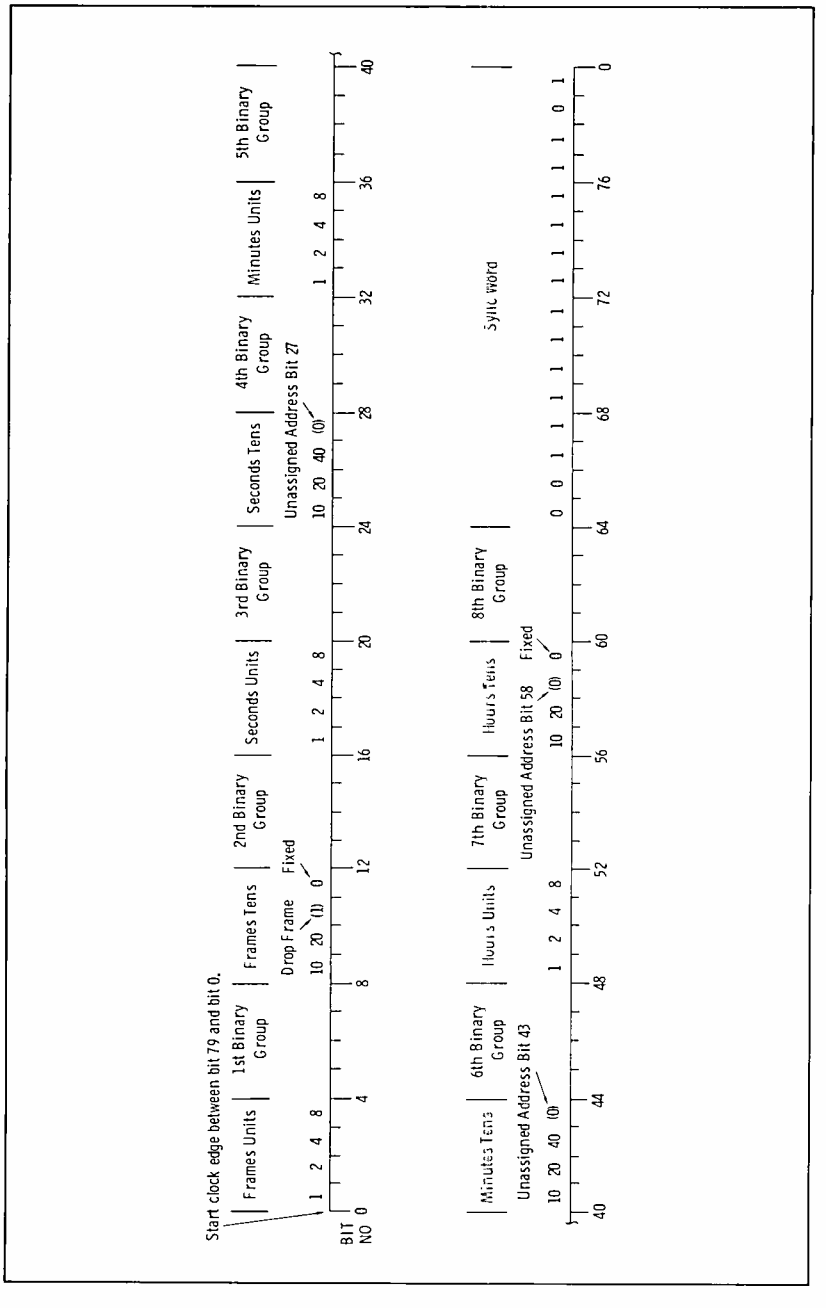


Fig. 5-8. SMPTE control time code.

with the use of the standardized SMPTE Video Tape and Audio Time Control code. This code uses a biphase mark, with binary-coded-decimal (bcd) time of day, bcd frame count, and optional binary word information.

The code, illustrated in Fig. 5-8, consists of 80 bits per frame, or 40 bits per field. Since there are 60 fields per second, the total number of bits per second is $60 \times 40 = 2400$. The code is normally recorded on the second audio (cue) track of a vtr, and the bandwidth is entirely adequate for normal forward tape speeds. However, in editing, it is convenient to use the code to search for required editing points during fast forward and reverse shuttling of the tape. This feature requires a bandwidth approaching 100 kHz.

Note from Fig. 5-8 that the SMPTE code consists of 32 user binary bits, 16 sync bits, 2 fixed zeros, 27 assigned addresses (including dropped frame, or DF, described below), and 3 unassigned addresses. All unassigned bits are zeros when not used. Thus, the total of 80 bits per frame contains data for hour, minute, second, and frame count; eight optional 4-bit words for additional information or control; and a sync word to indicate the end of each frame and the direction of the tape.

The sync word, bits 64 through 79, forms a special signal. The twelve 1 bits followed by a 0 and 1 indicate end of frame and forward direction of the tape. If the tape is in reverse, the twelve 1's are followed by 0 and 0, indicating reverse direction.

Fig. 5-9 shows an example of a biphase recorded SMPTE waveform for binary 10110110. The clock is at the beginning and end of each pulse. As shown, a 0 may be either "high" or "low." A 1 is distinguished from a 0 by the transition in voltage level at the midpoint of the bit time. The code is self-clocking, and it is immune to 180° phase reversals. Since the code waveform is rectangular, it can be recorded using either saturated-head or nonsaturated-head methods.

An example of the SMPTE code for the specific time of 16 hours, 47 minutes, 31 seconds, frame 23 is shown in Fig. 5-10. This wave-

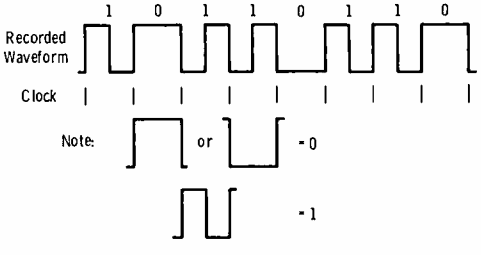


Fig. 5-9. Example of 10110110 in SMPTE code.

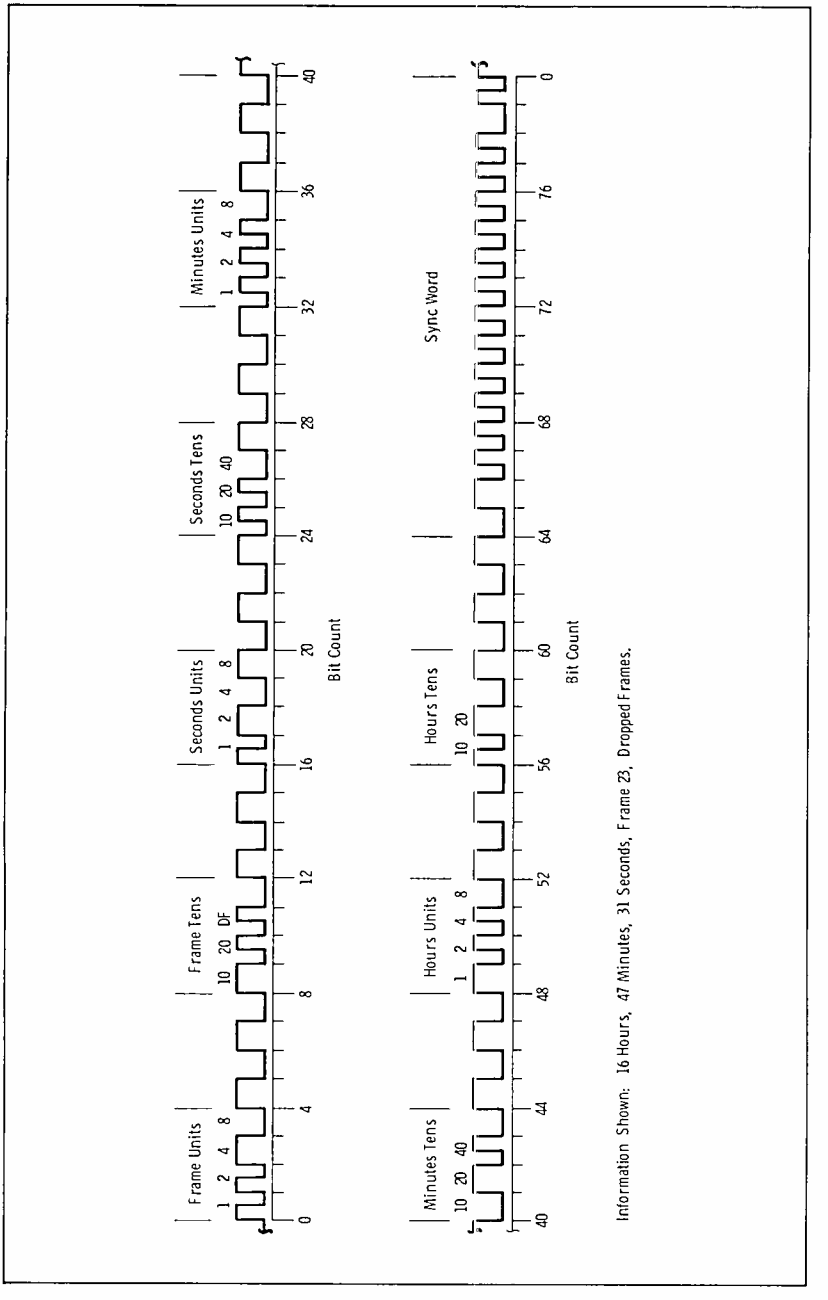


Fig. 5-10. Example of SMPTE code.

form also indicates that the dropped frame (DF, bit 10) is being used, since the tenth bit is a 1. We will now explore the necessity for this drop-frame technique.

Color video requires a *drop-frame* technique not required with monochrome recordings. The NTSC color frame rate is 29.97002 per second, instead of the monochrome rate of 30 per second. To prevent a resulting time error accumulation of 86 seconds per day, two frames are dropped every minute except every tenth minute. The inclusion of the DF bit permits either monochrome or color operation with the standard code. Examples of the use of this code in practice are included in Chapter 9.

5-7. EFFICIENCY OF CODING SCHEMES

The efficiency of any given coding scheme is based on the redundancy used in the code.

$$\text{Redundancy} = 1 - \frac{\text{Information bits}}{\text{Total bits}}$$

A 7-bit code using an eighth bit for parity would have:

$$\begin{aligned}\text{Redundancy} &= 1 - \frac{7}{8} \\ &= 1 - 0.875 \\ &= 0.125 \text{ or } 12.5\%\end{aligned}$$

A 7-bit code using three parity bits would have:

$$\begin{aligned}\text{Redundancy} &= 1 - \frac{4}{7} \\ &= 1 - 0.57 \\ &= 0.43 \text{ or } 43\%\end{aligned}$$

The efficiency (E) of any given coding scheme is:

$$\text{E} = 1 - \text{Redundancy}$$

Thus, the efficiency of the 8-bit code using 1-bit parity is $1 - 0.125 = 0.875 = 87.5\%$. The efficiency of the 7-bit code with three parity bits would be $1 - 0.43 = 0.57 = 57\%$.

Note that efficiency can also be obtained from the direct relationship:

$$\text{E} = \frac{\text{Information bits}}{\text{Total bits}}$$

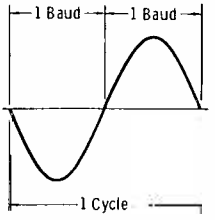


Fig. 5-11. One cycle contains two bauds.

All of the preceding simply indicates that the more the redundancy used for error detection, the less efficient the code. Error detection, or correction, is extremely important in digital communications.

5-8. INFORMATION CAPACITY

The amount of information in a message can be calculated by taking the product of the number of code elements and $\log_2 n$:

$$I = m \log_2 n$$

where,

I is the information capacity,
m is the number of code elements,
n is the number of possible values.

One cycle of a signal can contain a maximum of two bauds (Fig. 5-11). A baud can contain only one code element, but the code element can contain more than one bit if a more sophisticated coding is used. The binary element (one bit) has two possible values, 0 or 1. Thus, the information content is:

$$I = 1 \log_2 2 = 1 \times 1 = 1$$

NOTE: The value of $\log_2 2$ is 1 ($\log_x x = 1$). Values of $\log_2$ for other numbers may be obtained from Appendix C.

For a 4-bit binary code:

$$I = 4 \log_2 2 = 4 \times 1 = 4 \text{ bits of information}$$

But a ternary (base-3) code element can have any of three possible values, and a quaternary (base-4) element can have four possible values, etc.:

Binary (base 2) = 2 possible values (0 to 1).
Ternary (base 3) = 3 possible values (0 to 2).
Quaternary (base 4) = 4 possible values (0 to 3).

Thus, if a code consisted of four ternary elements, the information content would be:

$$I = 4 \log_2 3 = 4 \times 1.6 = 6.4 \text{ bits of information}$$

If a code consisted of four quaternary elements:

$$I = 4 \log_2 4 = 4 \times 2 = 8 \text{ bits of information}$$

This doubles the information capacity obtained from the straight binary code.

Dibits (pronounced "die-bits") provide a simple method of increasing the rate of bits through a given channel. One dibit is a pair of bits. There are four possible dibit combinations that may be assigned voltages, as in this example:

Dibit	*Voltage*
00	0
01	1
10	2
11	3

Assume you have the original bit stream shown at A in Fig. 5-12. Pairing of adjacent bits is shown at B, and the resulting dibits appear at C. Waveform D shows the conversion to the respective voltage values, halving the required baud rate of the original bit stream.

Each baud carries one dibit, and a dibit is a multilevel signal element, or baud. Each dibit is converted to its respective voltage level before transmission on the channel. Thus, bits are carried at twice the normal capacity of the channel.

Obviously, four tones of different frequencies could be (and are extensively) used instead of voltages. Bear in mind, however, that the increase in channel capacity obtained with the dibit code is not entirely free; terminal equipment is more complicated since it no longer has the simple job of easily detecting a bit. Thus, the decoder is more complicated.

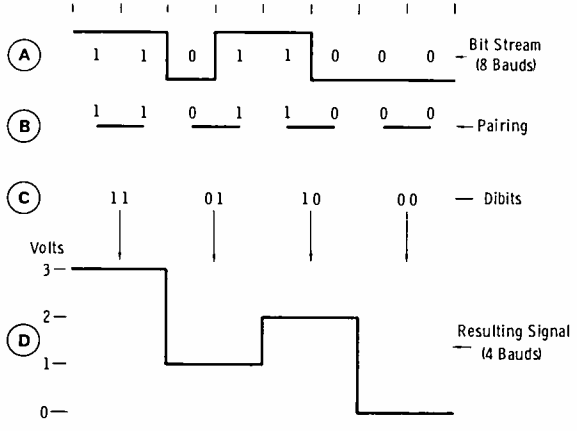


Fig. 5-12. Dibit encoding.

Another encoding form is to convert each dibit to one of four phases of an analog signal. For example:

Dibit	*Phase*
00	0°
01	90°
10	180°
11	270°

EXERCISES

Q5-1. A 5-2-2-1 code is specified.
(A) Is this a weighted or nonweighted code?
(B) What is the radix, if any, of this number system?

Q5-2. What is the advantage of any "two-out-of-five" code?

Q5-3. Using your knowledge of binary weights, construct the table for the 7-4-2-1 code, numbers 0 through 9.

Q5-4. Describe the difference between odd and even parity. What advantage (if any) does odd parity have over even parity?

Q5-5. What is the (A) redundancy and (B) efficiency of the bcd code?

Q5-6. What is the efficiency of a 4-bit binary word in which all four bits carry message information?

Q5-7. What is the efficiency of a 5-bit code that contains one parity bit?

Q5-8. If a code word of 14 bits is to be followed by its complement as a self-check, what is the efficiency of the code?

CHAPTER 6

Memory, Storage, and Peripheral Devices

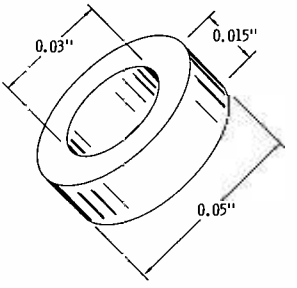
Basically, the terms *memory* and *storage* can be used interchangeably. However, the instructions for some systems may refer to an internal section such as a magnetic core as a memory device, and a physically separate magnetic disc or tape unit as a storage device.

The two oldest forms of memory are capacitors and relays. Capacitors can store a charge for a required period of time, then be discharged at a later time for readout. Relays can "remember" a given set of conditions in the form of the position of open and closed contacts.

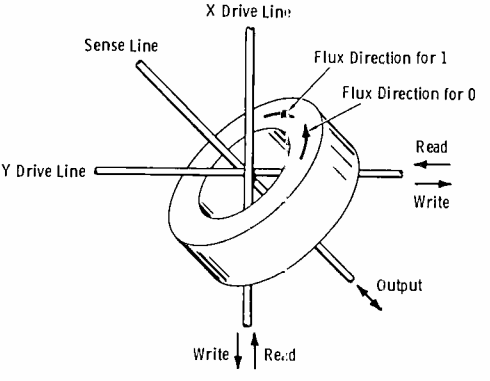
Modern digital systems normally employ flip-flops (ICs) or magnetic cores for internal memory. Magnetic discs, magnetic tape, punched paper tape, and punched cards are often used for external storage of information. This is about the finest line that can be used to differentiate between memory and storage. In practice, magnetic or punched paper tape and magnetic or punched paper cards are often called "programming" sources. See the Glossary (Appendix A) for basic definitions of types of memory and other terms.

6-1. MAGNETIC-CORE MEMORY

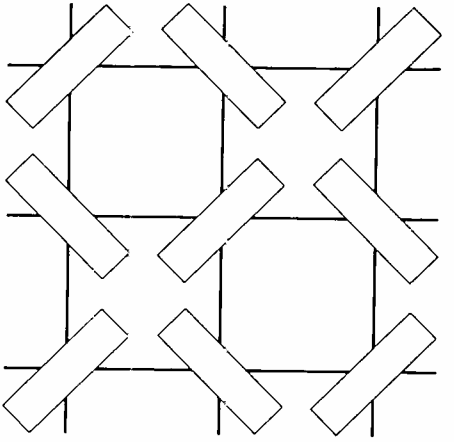
A magnetic core is a tiny ring of ferromagnetic material, with typical dimensions as shown in Fig. 6-1A. It can retain saturation in either of two states, which are arbitrarily assigned the values 0 and 1, depending on the direction of the current applied to the X and Y wires (Fig. 6-1B). Current through either X or Y alone is insufficient to "set" the magnetic core; both currents must be present simultaneously. A third wire senses the state of the core upon readout and provides the output signal.



(A) *Physical dimensions.*



(B) *Drive and sense lines.*



(C) 3×3 *array of cores.*

Fig. 6-1. Typical magnetic core.

A magnetic core provides one bit of information (0 or 1). Multiple cores are used for binary words. Fig. 6-1C illustrates the physical orientation for a 3×3 *array* to prevent mutual interaction of the individual magnetic fields. Typical dimensions for a 2600-bit (core) memory on a single plane ($\frac{5}{16}$ inch thick) are $4\frac{1}{2}$ by $4\frac{3}{4}$ inches. The cores are typically mounted on a printed-circuit board under a heavy sheet of clear plastic. The physical dimensions just given include the pc board and wires. These dimensions give an idea of the ultracompact package that can be achieved with magnetic-core memories.

Fig. 6-2 shows the basic operation of a magnetic core. The X and Y drive lines are two separate wires through the core, and they determine the direction of flux in the core. The write current places the core in the 1 state. The read current (opposite direction of write current) shifts the state of the core, and the sense wire picks up the signal voltage that results when the flux changes from 1 to 0. Note carefully that the X and Y wires are used for both the write and read operations. The write current direction "sets" the core to a 1; the read current "resets" the core to a 0. Thus, the information is destroyed during readout, and this type of memory is termed a *destructive-readout* (dro) type. If the stored information is to be used again, the read pulse must be followed immediately by a restore (rewrite) pulse to return the core to its original state. This is

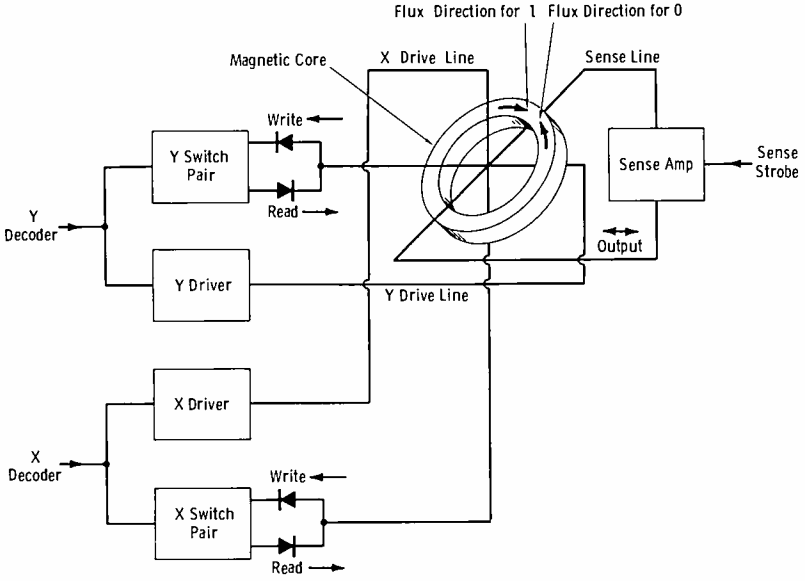


Fig. 6-2. Basic magnetic-core application.

termed a *nondestructive readout* (ndro), which will be described later.

The sense (output) pulse is quite small, typically around 50 millivolts. The sense amplifier shown in Fig. 6-2 is used to increase the signal-to-noise ratio. Noise pickup during the read cycle is further reduced by strobing. The amplified sense pulse is ANDed with an accurately timed pulse termed the *strobe pulse,* which occurs during the midamplitude portion of the sense pulse. This eliminates any output except that which occurs at the maximum energy level of the sense pulse (a change from 1 to 0).

Fig. 6-3 illustrates a simple 4 × 4 array on a single plane. Here we can see the purpose of the "X" and "Y" core-winding designations. In the example illustrated, simultaneous X and Y currents are being routed only to the core related to X4, Y2 on the rectangular-coordinate matrix. Although the X current is being routed to the other three cores in the X4 line, and the Y current also feeds the other three cores on the Y2 line, a single current alone will not change the state of the core. If simultaneous X and Y write currents are applied and core X4,Y2 is in a 0 state, it is shifted to a 1 state; if the core is already in a 1 state, no change takes place. If the core is in a 1 state and simultaneous read currents are applied, the core changes state (1 to 0), and the resultant voltage is picked up by the sense winding. Any core in the matrix is read in a similar manner by application of read pulses to the associated X and Y windings of the core.

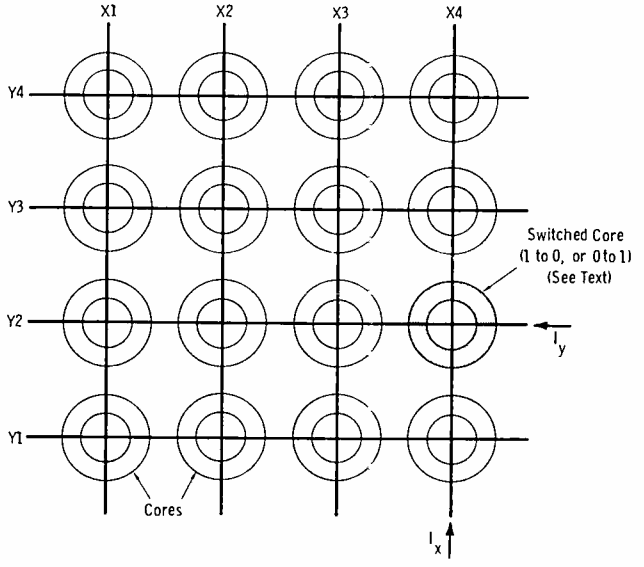


Fig. 6-3. Basic function of magnetic-core array.

If the core has not been switched to a 1 by a previous write current, it contains a 0. Application of a read pulse to the X and Y windings of the core does not switch the direction of magnetic intensity, and no pulse is induced in the sense winding. The read current can only switch a 1 to a 0, not vice-versa. Similarly, as previously stated, a write current can only switch a 0 to a 1. If a 1 already exists, no change occurs.

In some complex systems, a large number of "stacked" arrays (multiple planes) are necessary. The X and Y driver windings of each array are in series or parallel with every corresponding X and Y winding on other array assemblies. Some 1's and some 0's are normally written in one vertically stacked column of cores. When the X and Y wires are energized to write a 1, the intended cores in the stack must be selected. This is done by running an *inhibit* wire in parallel with either the X or Y line, adding a fourth winding to the core. A current is made to pass through the inhibit wire in the direction opposite that in the adjacent (X or Y) drive line. This cancels the magnetic field to prevent writing a 1 into any core that should be a 0.

The number of planes is determined by the number of bits in each binary word. Thus, for a 4-bit word, four planes are used. This technique is expanded in Section 6-2.

For nondestructive readout (ndro), all 1's stored prior to readout must be immediately rewritten into the appropriate cores. Fig. 6-4 shows the basic form of ndro for magnetic cores. The sense amplifier provides the 1's output either for direct use or for storage in an auxilliary memory register where final readout is made. The 1 is also fed back to the appropriate gate to supply an extra 1 from the initial write-pulse generator. In this example, the write generator is running at four times the timed write-pulse rate. For multiple planes, the inhibit wire is simultaneously energized from the opposite polarity of the flip-flop supplying the write pulse.

The *access time* (length of time between a call for data from storage and completion of the delivery) is very fast for magnetic-

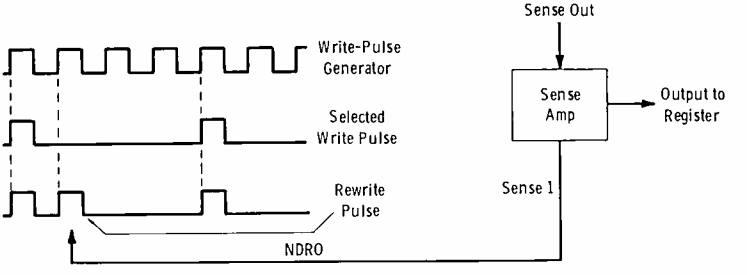


Fig. 6-4. Basic form of ndro for magnetic-core memory.

core memory. An access time of 1 to 5 microseconds is average, with some units running in the nanosecond range. Furthermore, the magnetic core has high retentivity and is a *nonvolatile* storage. This means that the binary information is retained even if power is removed from the unit. Core storage also provides a *random-access memory* (RAM), which means that the access time to the next desired storage location is independent of the previously addressed location. In contrast, magnetic tape has an output in sequence and is not normally a random-access device. However, it can be made to "search" for a desired location by allowing a sufficient time for the search operation. This time is in seconds or minutes rather than in the microsecond range of a true RAM, and it depends on the location of the desired data.

6-2. IC MEMORY (FLIP-FLOPS)

Integrated circuits consisting of multiple flip-flops have extremely fast access times and are inherently nondestructive-readout devices. Read access times of from 20 to 200 nanoseconds are typical. The ndro function results from the fact that the output sensing does not change the state of a given flip-flop, and no rewrite pulse is required.

Each flip-flop in an array is normally termed a *cell*. The arrays are stacked on top of each other as shown in Fig. 6-5, with the first bit stored in the top array, the second bit in the next to the top array, etc. To permit storing the bits of a word in the same relative cell position in the stacked arrays, the corresponding X and Y driver wires of each array are connected in series or parallel. Therefore, when the coordinate wires of one cell in an array are energized, every corresponding cell in the other arrays is also energized. The function of each cell is then dependent on the condition of the "write 1" (W1) or "write 0" (W0) inputs. This will become more evident as we continue.

In the example of Fig. 6-5, a basic chip contains 16 bits. A single array (plane) consists of four such chips, or $16 \times 4 = 64$ words of one bit each. Thus, four planes result in a 64-word–by–4-bit memory system. Planes may also be "stacked" side by side.

Information on the write inputs can be written into four bit locations simultaneously. Four bit locations can also be read simultaneously. In the latter case, the "sense 1" output (S1) will be the OR of the 1's stored in the four bit locations, and the "sense 0" output (S0) will be the OR of the 0's stored. (Some diagrams show a "wired OR" symbol.) Thus, when four bits are read simultaneously, the "1" and "0" outputs of the array *may not be complements.*

In practice, multiple-emitter transistors are used in flip-flops (Fig. 6-6). Before analyzing Fig. 6-6, you should become familiar with

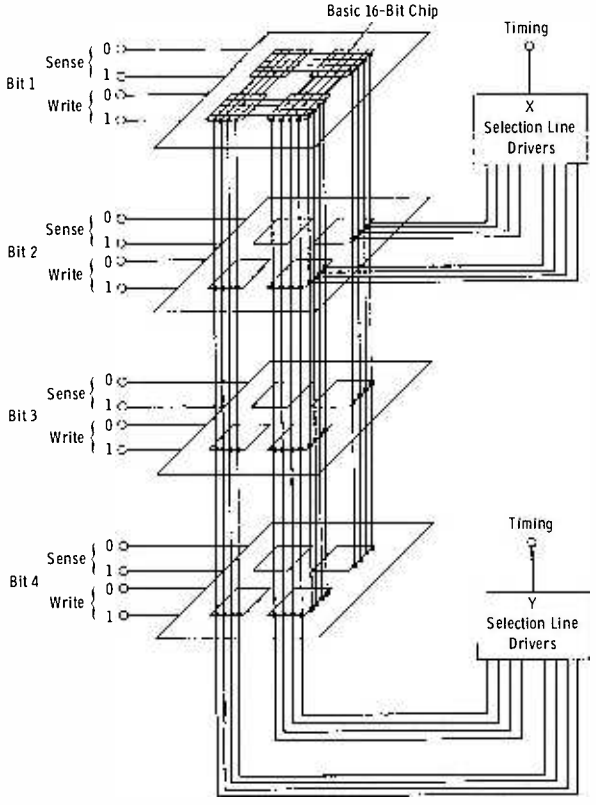


Fig. 6-5. Example of 64-word, 4-bit memory.

typical pulse nomenclature as applied to memory circuits by noting Fig. 6-7. The important characteristics are listed and illustrated in this diagram and will be referred to during the following discussion.

Fig. 6-6 is an example of one flip-flop (cell) in the basic chip. For the example of Fig. 6-5, this is one of the 16 total flip-flops of a chip. The W1 and W0 (inverter) amplifiers and the S1 and S0 amplifiers are common to each array. The write amplifiers provide the proper fan-out capabilities to the other flip-flops in the same matrix array. The fan-in for the sense amplifiers is the total number of flip-flops in the array.

Basic Write Operation

Assume Q1 in Fig. 6-6 is on, Q2 is off, and the X and Y inputs for the bit time are high. To write a 1, the write 1 input (W1) is driven high. This opens D1, and current is routed to Q3 and Q4

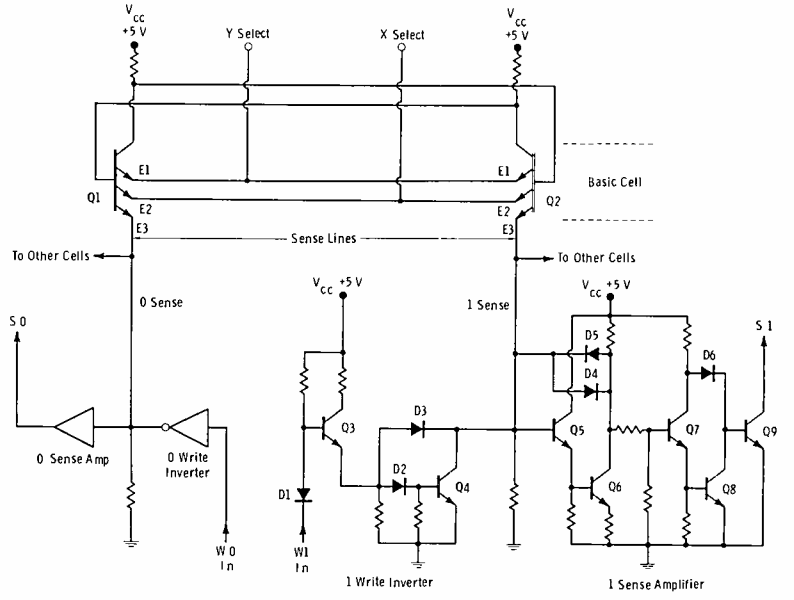


Fig. 6-6. Simplified example of memory cell with associated write and sense amplifiers.

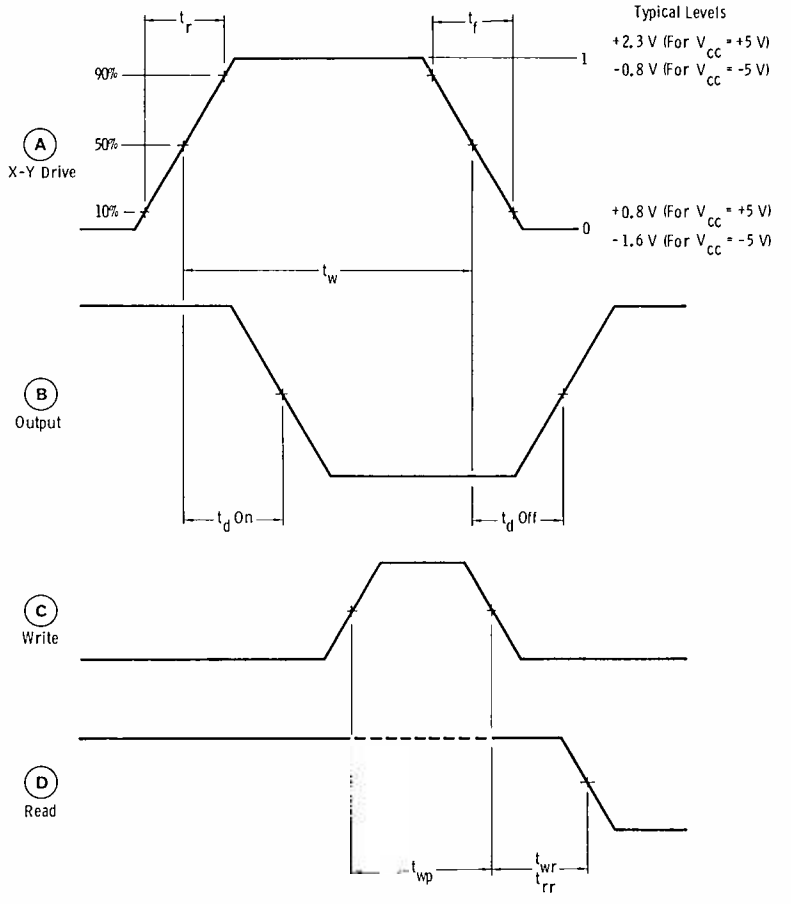
from V_{CC}. Thus, the Q4 collector voltage drops to the level of the base-emitter voltage (low level). Diode D3 prevents full saturation of Q4 in the interest of maximum write and recovery speed.

Since the collector of Q4 is now close to ground (approximately +0.4 volt), Q2 is driven from cutoff to conduction, Q1 is cut off, and the state of the flip-flop is changed. The sequence of operation may be outlined as follows:

1. Prior to the write pulse, the X and Y selected lines are low (less than +0.8 volt).
2. When the bit in question is selected, the X and Y lines are driven high (over 2 volts). See Fig. 6-7A.
3. A 1 is written by driving W1 high (over 2 volts) for a minimum time (t_{wp}, minimum write-pulse width) as specified for a given chip. See Fig. 6-7C.
4. The W1 line is returned to the low state.
5. The stored bit can be read only after the write recover time (t_{wr}). See Fig. 6-7D. The sense output during the write time is indeterminate.

Basic Read Operation

The sense line (emitter 3 of Q1 and Q2) is biased at approximately +1.5 volts from the sense amplifier when there is no current



t_w = Width of Pulse Measured Between 50% Amplitude Points
t_r = Rise Time Measured Between 10% and 90% Points
t_f = Fall Time Measured Between 90% and 10% Points
t_d On = Turn - On Delay Time
t_d Off = Turn - Off Delay Time
t_{wp} = Write Pulse Width
t_{wr} = Write Recover Time
t_{rr} = Read Recover Time

Fig. 6-7. Basic pulse nomenclature.

from the memory cells in the associated sense line. The collector voltage at Q6 turns Q7 and Q8 on, causing Q9 to be off (nonconducting). When selection occurs, the resulting sense-line current increases the base drive to Q5 and Q6, causing the Q6 collector voltage to lower. Transistors Q7 and Q8 are turned off, causing Q9 to turn

on. Therefore, when a 1 is stored in the selected bit, the S1 output goes low. When a 0 is stored, the S0 output goes low.

The read-operation sequence may be outlined as follows:

1. Assume a 1 is stored in the selected bit. The X and Y selection lines and both write inputs are low (less than +0.8 volt).
2. The desired bit is selected by driving the X and Y selected lines more positive than 2 volts (Fig. 6-7A).
3. See Fig. 6-7B. After the turn-on delay time (t_d On), the S1 output will be low (less than +0.4 volts) and the S0 output will be high (more than 2 volts). Note that the read operation does not change the state of the flip-flop, and memory is retained until a change is caused by subsequent write operations.

NOTE: A recent development in the IC field is the availability of "4K" single RAM units with a capacity of 4096 bits. These are found in both 22-pin and 16-pin configurations. Access speeds of 100 to 150 nanoseconds are typical.

Read-Only Memory (ROM)

A read-only memory (abbreviated ROM) is an IC chip from which digital data can be read out over and over (ndro), but which cannot be written into without special processing. There are three general classifications of ROM:

1. The basic ROM is programmed by the manufacturer as the IC chip is fabricated. Such chips are available in a wide variety of functions such as code conversion or highly complex truth tables that meet user requirements.
2. The programmable ROM (PROM) can be programmed by the user at his own location using a special type of programming equipment that reads data into the memory of the ROM. Once programmed, it cannot be altered. Manufacturers of PROMs will program to meet your needs if you supply the required truth table.
3. With the erasable PROM, the memory can be repeatedly reprogrammed. Older types required ultraviolet light to erase the old program. Later types require a single pulse to erase the entire memory of data. Special equipment is still required to reprogram, unless you have the manufacturer do this for you.

6-3. MAGNETIC RECORDING FOR PERIPHERAL DEVICES

The memory and storage techniques discussed thus far are typical of "internal" circuitry. Since the operation is all-electronic, the speed is high, making possible thousands or millions of operations

per second. However, this internal action of the basic digital unit must be interfaced with the operator. Individuals must prepare the data to be used by the basic digital unit, and this unit must in turn present its results for use by people. This interfacing is termed input-output (i/o) bussing. *Peripheral equipment* (those devices employed external to the digital system) requires the use of materials that must be handled physically and moved mechanically at speeds much slower than the electronic speeds within the basic digital unit. Mechanical speeds are measured in terms of seconds or milliseconds. Speeds within the basic digital unit are measured in terms of microseconds or nanoseconds.

The recording of digital information on a magnetic surface (whether card, tape, disc, or drum) is essentially the same as audio or video tape techniques, with which the reader should already be familiar. There are, however, special considerations which will be discussed in this section. (Further mechanical requirements are covered in Section 6-5.)

In digital magnetic recording, the magnetic surface may touch the pole faces of the head, or it may be separated from them by a very thin airstream of less than one-thousandth of an inch. The magnetic read-write core has a very fine gap much smaller than a thousandth of an inch.

There are a variety of recording techniques for digital information, including return to zero (RZ), return to bias (RB), and non-return to zero (NRZ). The method employed depends on several factors, such as recording density, amount of information to be recorded, type of timing signals to be used, and method of erasing.

Return-to-Zero Method

In RZ recording, current is passed through the head winding in one direction to write a 1 and in the opposite direction to write a 0. In this method, an erasing head must be used to clear the previously recorded signal. This erase feature may be included in the assembly, or an external bulk eraser may be required.

Return-to-Bias Method

With the RB method, a "bias" current is maintained through the head windings to saturate the magnetic medium in the direction of erasure. When a 1 is recorded, a pulse of recording current is sent through the head, resulting in a spot of magnetization opposite to the erase condition. If no 1's are furnished to the recording head, the medium will be completely erased (recorded in the zero state), leaving all 0's. The resulting signal output is essentially the same as with RZ recording, but it is not necessary for the medium to be previously erased.

Nonreturn-to-Zero Method

The NRZ method of recording is another method that does not require the medium to be erased before new information is recorded. Observe Fig. 6-8 and note that each time a 1 is recorded, the head recording current (hence recorded magnetization) is caused to reverse. Each reversal results in a read-back signal. Zeros produce no transitions, hence no read-back signals. Thus, the short pulses of previous methods are not used, and the current never returns to zero or to a bias condition.

Note carefully that the NRZ method of magnetic recording is *not* identical to the SMPTE Timing Code (Section 5-6). With the SMPTE code, there is at least one transition for every bit. This makes it possible to derive a clock or timing signal from the information channel itself. Other methods require a separate timing channel.

6-4. THE MAGNETIC DISC

A magnetic disc is an aluminum disc resembling a phonograph record. Sizes vary considerably, depending upon application. Bonded to both sides of the disc is a ferromagnetic oxide suspended in an

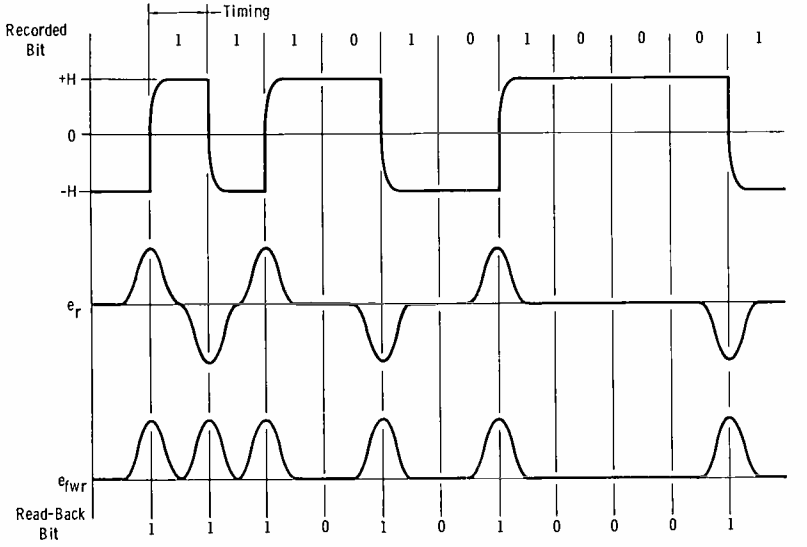
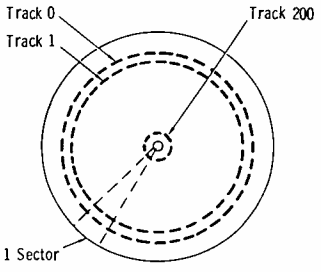
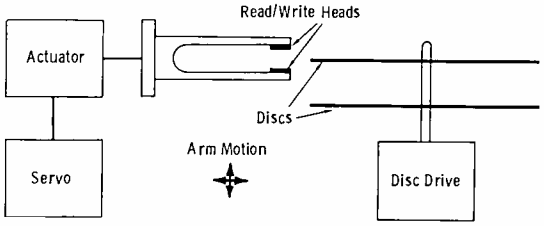


Fig. 6-8. Basics of NRZ magnetic recording.



(A) *View of disc surface.*



(B) *Read/write mechanism.*

Fig. 6-9. Principle of magnetic disc for data storage.

organic binder. The iron oxide has magnetic characteristics similar to those of magnetic tape. Note that both sides of the disc are normally used, requiring a top and bottom read-write head (described later). The magnetic disc is widely used in peripheral data-storage applications.

A typical magnetic disc is represented in Fig. 6-9A. In this example, each side contains 200 tracks, for a total disc capacity of 400 tracks. In practice, 400 (and more) tracks may be used on a single disc surface. A *sector* is the smallest addressable block of data. This is determined by the specific system, the speed of rotation, and the information retrieval rate. Sectors may vary from 10 to 24 or more per disc surface.

There are two basic types of disc systems:

1. Fixed-head-per-track drives, which have multiple read/write heads for each disc surface.
2. Moving-head, interchangeable-cartridge type, which has only one head per disc surface (Fig. 6-9B). This is the most common type in digital broadcast system applications.

Fig. 6-9B is a simplified view of a two-disc, movable-arm system. In practice, stacks of from 1 to 20 or more discs may be used. The arm is moved out, down, and in when going from the top disc to the botton; it is moved out, up, and in when moving from the bottom disc to the top. In practice, this time is quite rapid, being mea-

sured in small fractions of a second. Some systems employ one arm for each disc in the stack. The engagement to any given sector position of a disc is measured in nanoseconds.

The head (two on each arm, one each for top and bottom) consists of a magnetic core which has coils of wire wrapped around it. The small gap in the core is positioned directly over the disc without contact. This spacing varies between 0.001 inch (maximum) and 50 to 90 microinches. Otherwise, the magnetic action is the same as in audio or video tape recording. The tracks on which data are recorded are typically spaced 0.01 inch apart, when the spacing of the head gap above the disc is around 70 microinches. To guarantee interchangeability, the heads must be located within ±0.0015 inch of the nominal track location, requiring tight servo-controlled positioning systems.

The same head is used for read or write, depending on the mode of operation. When used, a separate erase winding on the head allows a given portion to be erased when desired.

Airflow

Airflow is essential to a disc drive. The small spacing between the read/write head and the disc surface is maintained by a differential airflow over and under the head. The arm-carriage location errors caused by thermal offsets are minimized by passing a large quantity of cooling air through the disc drive.

The air must be cleaned before it is introduced into the disc system. The heads fly over the disc at a typical height of 65 to 90 microinches, and any small particle of dirt can scratch the disc or head. A single scratch can cause a snowballing effect that can ruin the head and the disc. Two filters are normally used: a coarse filter behind the main air intake, and a fine filter on the high-pressure side of a squirrel-cage blower. The fine filter traps all particles 12.5 microinches and larger.

Data Handling

An an example, a sector (smallest addressable block of data) might contain a zero field (string of zeros), a sync word, sector verification, a 256-bit data field, and a cyclic error-check word. The field of zeros at the start of a sector is used to synchronize the data-separation circuitry. In read operations, a control loop locks onto the zero field in time to be operating at the proper read clock rate for data separation.

As data are written, they are checked cyclically, and a cyclic check word is formulated. This word is written after the data field in each sector. When the sector is read, it is again checked cyclically, and the results are used to determine if an error occurred during the

read operation. The sync word and sector verification allow tight control over signal certification. These are not always used in some simpler systems.

Typically, three levels of addressing specify the location of data on the disc. Radial location may be specified by an 8-bit track address, angular location with respect to a reference may be specified by a 5-bit address, and the disc surface (top or bottom may be specified by a 2-bit address.

If there are 24 sectors at 256 bits per sector:

$$(24)(256) = 6144 \text{ bits per track}$$

If there are 200 tracks per side:

$$(200)(6144) = 1{,}228{,}800 \text{ bits per side}$$

Counting both sides, the total capacity for each disc is:

$$(1{,}228{,}800)(2) = 2{,}457{,}600 \text{ bits/disc}$$

Thus, a single disc in the above example has a capacity of close to 2.5 million bits.

The exact data operation varies with the type and manufacturer of the magnetic disc system. For example, the Hewlett-Packard Type 7900A is used to transfer a 1.25-million-word record at an average data transfer rate of 82,000 words per second. First, sectors 0 through 15 on track 0 are transferred, and the heads are moved to track 1. This requires 16.7 milliseconds for data transfer and 7 ms for seek time (head positioning), for a total of 23.7 ms. This is less than the time of one disc revolution, which is 25 ms (more fully discussed below under "Access Time"). After a 1.3-ms wait, sectors 0 through 15 on track 1 are transferred, and the heads are moved to track 2. This is repeated for all 200 tracks. The process is then repeated in reverse, transferring sectors 16 through 23. The system then switches heads (this takes only a few microseconds) and transfers sectors 0 through 7, then 8 through 23 on the other disc surface.

Access Time

Maximum access time is the time needed to position the head mechanism plus one disc revolution. Disc rotation is typically between 1200 and 3600 rpm. For example, at 2400 rpm:

$$2400/60 = 40 \text{ revolutions per second}$$
$$1/40 = 0.025 \text{ second per revolution}$$
$$= 25 \text{ ms/revolution}$$

This time of 25 ms is termed *latency*, or rotational delay. It is the time required for the disc to rotate so that the desired data are

beneath the heads. The *maximum* time would be one complete revolution. An *average* delay may be taken as one-half revolution, or $25/2 = 12.5$ ms.

Seek time is the time required to position the read/write heads over the desired track on the disc. When only a single disc is used, or when each disc has a positioning head assembly, the seek time can be very short, typically less than 0.02 second as a maximum.

Since the access time is the sum of seek time and latency, the maximum access time in this example is:

$$25 \text{ ms} + 20 \text{ ms} = 45 \text{ ms} = 0.045 \text{ s}$$

6-5. OTHER MAGNETIC STORAGE DEVICES

Magnetic discs (described above), drums, cards, and tape provide the primary peripheral storage facilities for basic digital systems. All of them operate on the magnetic principles described in Section 6-3.

The Magnetic Drum

The drum is a cylinder with a magnetic coating; it rotates past a large number of heads. On earlier drums, the heads were rigidly mounted to the housing and adjusted to within one-thousandth of an inch above the drum surface. More recently, heads are mounted on small spring-loaded pads that "float" on an airstream less than one ten-thousandth of an inch thick.

A drum 10 inches in diameter might have 15,000 bits around the circumference. With 256 heads, this drum would have a capacity approaching 4 million bits. Drum speeds vary widely, ranging from 1800 to 18,000 rpm. At a speed of 3600 rpm, one revolution takes $\frac{1}{60}$ s, or 16.67 ms. This would be the maximum access time for the drum.

All bits stored are not information bits, and this is common to all systems. If there are 15,000 bits around the drum, one channel may have 15,000 pulses recorded in it for "framing" the readout. Another channel may have a single pulse recorded for each revolution. Another channel may have numbers recorded in it to serve as "addresses" for words stored around the drum.

The Magnetic Card

A magnetic card is made of oxide-coated plastic that is considerably thicker than magnetic tape in order to provide stiffness. The cards vary from 3 to 5 inches wide, and they are 12 to 16 inches long. The deck of cards is held in a cartridge in such a way that the desired card can be selected by means of coded notches along

the edge. Each card has a different code, and the selection scheme can sort it out without reference to location in the deck.

When a card is selected, it is moved along a chute by an air-stream or by rollers until it emerges on a rotating drum equipped with a number of read/write heads. The card is held on the drum surface by a vacuum or other means, with the magnetic surface facing outward.

When read or write has been completed, the card is released from the drum and travels by way of a return loop to the top of the deck in the cartridge. It need not be returned to a particular location in the deck because the selection system can find it on demand by means of the coded notches. A new card can be selected and secured to the drum in a fraction of a second.

A typical cartridge holds 256 cards, each of which holds around 3100 alphanumeric characters. Typical access time is under 0.5 second.

Magnetic Tape

As mentioned previously, it is assumed that the reader is familiar with basic magnetic tape recording. The basic electronic differences for digital applications were covered in Section 6-3. There are also a number of mechanical differences which will be covered in this section.

Magnetic tape for digital use may vary from ½ to 3 inches in width, and there may be up to 3600 feet of tape on a single roll. The size most frequently used is ½ inch by 2400 feet.

Digital information is arranged *across the width* of the tape, usually with seven to nine channels. This means that the read/write head is capable of reading or writing seven to nine bits across the tape width. The tape travels at speeds ranging from 50 to 200 inches per second. The number of bits that can be recorded per inch of tape depends on the head-gap size and the character of the magnetic surface, and it ranges from 200 to 1000 bits/inch. This is termed the *recording density*. Thus, for a tape speed of 100 in/s, 9 channels, and a recording density of 800 bits/in, the data rate is:

$$(100)(9)(800) = 720{,}000 \text{ bits/s}$$

Data are usually organized in the form of "records." A record may occupy only 1 or 2 inches of tape. At a density of 800 bits/in, a 2-inch length would contain 1600 bits per channel. With seven channels, the data might be arranged in 6-bit characters across the width of the tape, leaving the seventh channel for purposes of timing or error control. Thus, 1600 six-bit characters are provided. Assuming a basic digital unit employing 24-bit words, this number of characters represents 400 digital words ($1600 \times 6 = 9600$, and

9600/24 = 400). This illustrates how a large digital word of a basic digital unit may be stored in smaller bytes on a peripheral device.

The above example of a record occupying only 2 inches of tape also emphasizes another important point in digital tape operation. Information is generally exchanged with the basic digital unit in short bursts. This requires the tape to start instantly, move at constant speed for a short distance, and then stop abruptly.

A full reel of magnetic tape is much too heavy to be started and stopped for only a few inches of tape by conventional means. The problem is solved by providing loops of slack for the supply reel and the take-up reel. In high-speed units, the slack is formed by allowing loops of tape to hang into vacuum chambers, one after the supply reel and another following the capstan and read/write head assembly. The capstan is reversible; it allows tape motion either forward or backward. The vacuum chambers maintain tension in the tape while allowing slack so that the tape can start moving instantly. When movement starts, the slack loop lengthens on the take-up side and shortens on the supply side. ("Take-up" and "supply" depend on the direction of motion.) As the lengths change, the loops move past sensing devices (optoelectronic or pressure) that cause the reel servo motors to start operating. The reels turn as much as required to restore the slack loops to their nominal lengths.

In general, a vacuum capstan rather than the pinch-roller type is used to obtain the fastest possible response. The rotating capstan either is of porous material or has many holes drilled around it. The capstan is connected to a chamber that can supply either pressure or vacuum in response to a control signal. With no tape motion, the capstan is rotating, but air pressure holds the tape slightly away from its surface. For tape motion, vacuum is applied, drawing the tape against the rotating capstan and bringing it instantly to full speed.

When the tape is stopped, the capstan is again pressurized, driving the tape away from its surface. Simultaneously, vacuum is applied to a stationary surface adjacent to the capstan to brake the tape to an instant stop.

A typical high-speed tape unit is able to bring the tape up to full speed in 1 millisecond and stop it in 2 milliseconds. During the start and stop intervals, the tape motion changes rapidly, and information cannot be read or written. These intervals are called *interrecord gaps* (also termed *data blocks*). At an average speed of 100 inches per second, the tape would move 0.3 inch in the total start-stop interval (1 + 2 = 3 ms, and 0.003 s × 100 in/s = 0.3 inch).

It is customary to leave about 0.6 inch for interrecord gap. This ensures adequate distance for starting and stopping, and allows for

uncontrolled variations. Thus, if an entire reel of tape were filled with 2-inch records and 0.6-inch interrecord gaps, the portion of the tape that would not be used is $0.6/2.6 = 23\%$.

NOTE: There are many recent developments in magnetic-medium data equipment. These include very flexible "floppy" discs (one-sided), "flippy" discs (two-sided), ¼-inch data cartridges, and minicassettes.

The minicassette is the latest development at the time of the preparation of this book. It is particularly applicable to data storage in hand-held and desk-top digital devices. It typically packs 50 feet of recording medium (64,000 characters at 600 bits per inch) in a package smaller than a box of pocket matches ($2 \times 1.3 \times 0.3$ inches).

6-6. THE ELECTRIC TYPEWRITER

The electric typewriter is an electromechanical device. Energy for operation is obtained from an electric motor that operates continuously. The typewriter is generally intended to operate as an input and output device. For use as an input, it produces coded signals that are sent to the digital control unit. As an output, the typewriter is operated by coded signals from the digital control unit, generally at speeds from 10 to 20 characters per second.

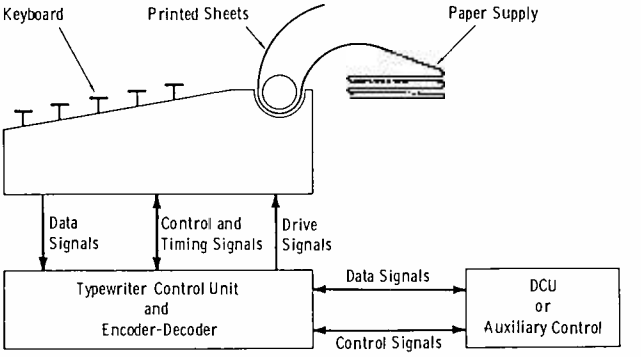
One of the most common applications in broadcasting is the use of the keyboard (input) to enter a program into an auxiliary or buffer storage for the main digital control unit (dcu). Changes or additions can be made at any time. A different keyboard (output) may be used for automatic logging of aired programs and spots.

Fig. 6-10A is a simplified presentation of an application of an electric typewriter. When a character is struck on the keyboard, the typewriter control unit converts this to a series of binary pulses, which may be pure binary, octal, bcd, or any other coded format. These data are stored in the appropriate memory unit. When the typewriter is operated as an output device, the data are decoded and actuate the appropriate key for the character represented by the data (Fig. 6-10B).

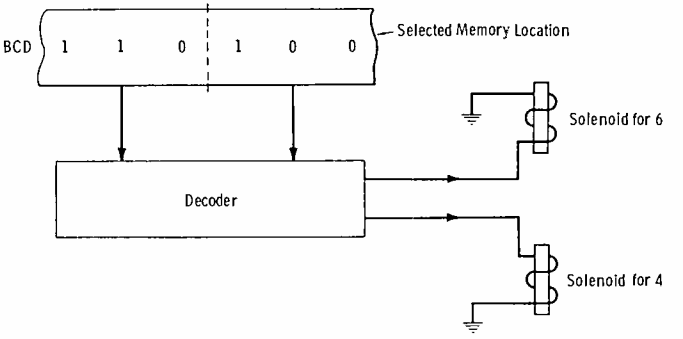
NOTE: There are many other forms of printers used in conjunction with main computer operations. These include chain printers, high-speed line printers, electrostatic printers, and photographic printers. These are not widely used in communications applications and are not covered in this book.

6-7. PUNCHED CARDS AND PAPER TAPE

Thin cardboard cards and paper tape may be punched to represent any data that need to be stored. A single standard punched



(A) Example of function.



(B) Readout of bcd 110 100.

Fig. 6-10. Electric typewriter as input and output devices.

card is 80 to 90 columns in width and around 13 rows in height. Numbers, letters of the alphabet, and various symbols are represented by the relative locations of small rectangular holes punched in the cards with a "key punch." Eighty or ninety alphanumeric characters are stored on each card.

Blank cards are stored in a hopper on the card-punch machine. They are moved one at a time under the rectangular mechanical punch knives, which are operated by solenoids similar to those of

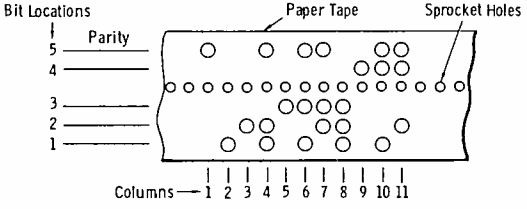


Fig. 6-11. Example of punched paper tape.

the electric typewriter. The speed of card punches varies from 150 to 500 cards per minute.

On paper tape, columns of small circular holes are punched across the tape as indicated in Fig. 6-11. In this example, a 4-bit binary code is used, with a fifth position for parity. Each circle indicates a hole; note, therefore, that *odd parity* is used. Generally a hole represents a 1, and a blank space represents a 0. Thus, in Fig. 6-11, the word in column 1 is 00001, the word in column 2 is 10000, the word in column 3 is 01000, etc.

Both cards and tape are normally read out by optoelectronic devices for maximum speed. Although the punching of holes in paper tape may be relatively slow (50 to 300 characters per second), electronic readout techniques allow a read speed of better than 1000 characters per second.

EXERCISES

Q6-1. What is the purpose of the sense wire in a magnetic core memory?

Q6-2. What is the purpose of a strobe pulse?

Q6-3. What is the storage capacity (in bits) of a 16×16 core array?

Q6-4. What determines the matrix capacity and the number of planes in a memory?

Q6-5. Define: (A) dro, (B) ndro, (C) RAM, (D) volatile storage, (E) nonvolatile storage.

Q6-6. Is it necessary to erase a magnetic medium before new data can be recorded on it?

Q6-7. What factors determine the density of magnetic recording?

Q6-8. Must peripheral devices use the same size words as the basic digital system?

Q6-9. A magnetic disc rotates at 3600 rpm and has a maximum arm-positioning time of 0.1 second. What is the maximum access time?

Q6-10. What advantage do you see in using magnetic cards instead of magnetic tape?

CHAPTER 7

A/D and D/A Conversion Techniques

Analog-to-digital (a/d) conversion is called *coding* or *encoding*. Digital-to-analog (d/a) conversion is termed *decoding*. Analog-to-digital conversion is often abbreviated *adc*. Similarly, digital-to-analog conversion may be abbreviated *dac*. The combination of coder-decoder is termed a *codec*.

There are two general forms of analog-to-digital conversion systems:

1. The adc direct method with no intermediate steps. This includes open-loop and closed-loop methods.
2. The adc with an intermediate step, which can include dac within a closed loop. This becomes a closed-loop codec.

Both general categories will be covered in this chapter, with emphasis on the methods most frequently encountered by broadcast and communications technicians. Only the basic theory of operation will be covered in this chapter. More detailed circuitry is covered for specific equipment in Chapters 9 and 10.

7-1. FUNDAMENTALS OF CODEC THEORY

The codec (coder-decoder) system is the heart of all digital systems. The faithfulness of transmission and reproduction (resolution) and the signal-to-noise (s/n) ratio for a digital system are largely determined by the design and operation of the codec.

Assume an analog signal is applied to the input of the network in Fig. 7-1. The source resistance (R_s) is low. The impedances of the

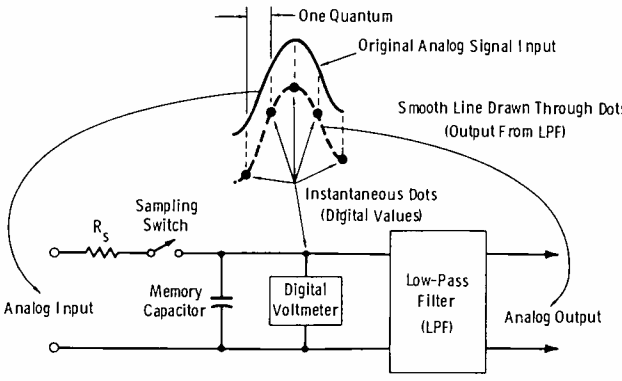


Fig. 7-1. A simple codec.

voltmeter and low-pass filter in this simplified example are very high. The momentary-contact switch connects the storage capacitor to the analog signal for a very short time as indicated by the dots on the drawing. Thus, the capacitor is charged (or discharged) to the new voltage value of the analog signal at the instant of switch closure. At this point, the analog signal has been converted (coded or encoded) to a series of instantaneous voltage values. The low-pass filter is equivalent to connecting the dots with a smoothly varying line, and is the decoder in this simple example.

It should be understood that a digital representation of an analog signal is never a 100% true equivalent. The frequency of the "sampling" (switch operation) once and for all determines the system *resolution,* as well as the s/n ratio. The period between samples is termed a *quantum.* The smaller the quantum (the higher the sampling rate), the less the chance of missing peaks or valleys in a rapidly changing waveform.

In the example of Fig. 7-1, the digital system is capable of recognizing four different levels. This can be represented by two digits ($2^2 = 4$). Thus, a quantum is one part in four. If a 4-bit binary is used, the system will recognize $2^4 = 16$ levels, and a quantum will be one part in sixteen. An 8-bit binary will recognize $2^8 = 256$ levels, with each quantum one part in 256.

There are many varieties of codecs, each with certain advantages and disadvantages for specific applications. The digital frequency meter or ac voltmeter is a relatively simple codec which converts the analog signal to a digital readout. Digital control is often a simple and stable timing system that merely controls analog signals without converting analog signals to digital signals. Digital audio and video, used in time-base correctors, automatic synchronizers, special effects, etc., are examples of exacting requirements in dac and adc.

7-2. THE BASIC DIGITAL FREQUENCY METER

Because of its simplicity, the analog-to-frequency converter is one of the most common open-loop encoders. Essentially, it counts the number of cycles per unit time. The unit time is determined by a known reference clock frequency. This application is illustrated in Fig. 7-2.

Recall that a trigger to the S input sets the scaler to a binary 1 state, represented by a plus voltage on the 1 output but not on the 0 output. A trigger to the R input switches the scaler to the binary 0 state, represented by a plus signal at the 0 output only. Also, a trigger to the T input toggles the flip-flop to reverse whatever state it is in. (Review Section 4-6 and Fig. 4-10 for basic counter action.) The added elements in Fig. 7-2 are the reference clock frequency, the divider, an enable flip-flop, and an AND circuit. The AND circuit will produce an output only when both inputs are high.

The divider multivibrator output is essentially delayed from the clock signal so that the binary counter can be reset to zero immediately prior to the beginning of each counting interval. In this way, the number that appears following the stop pulse represents only the total count accumulated between start and stop pulses from the divider multivibrator. The start pulse triggers the enable flip-flop to the binary 1 state, which allows the signal being measured to pass through the AND circuit until a stop pulse occurs. This is the

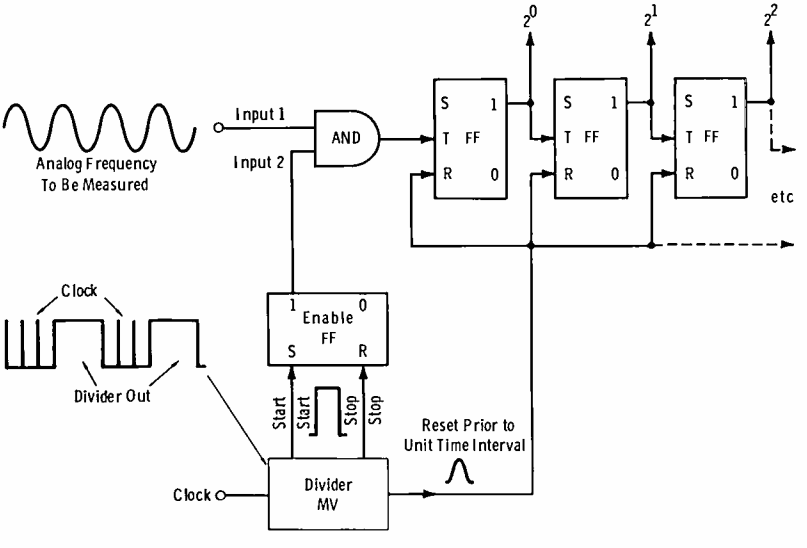


Fig. 7-2. Basic digital frequency meter.

reference unit time. The stop pulse triggers the enable flip-flop to the binary 0 state.

Note that the binary counter of Fig. 7-2 is a simple cascaded series of binary scalers that count in the natural binary code. They can also be connected in such a way as to count in the decimal number system, a bcd code, the octal code, or any other code, by the use of suitable decoders.

The accuracy of the frequency counter depends largely on the accuracy and stability of the clock signal, which may be from a crystal-controlled oscillator of 100 kHz or another reference frequency. Note that the width of the divider pulse determines the *sampling period*, hence the number of cycles counted. The start of the count might occur at any point in the first cycle of the measured signal, so an additional error of plus or minus one count can occur. This error, expressed as a percentage, can be minimized by increasing the sampling period at the expense of a greater time between successive readings.

7-3. THE BASIC COMPARATOR

Many adc techniques are based on the *comparison* method. Therefore, it is necessary to understand the basic form of comparator. Although the reader is expected to have a background in basic transistor and IC applications, we will review the differential amplifier because of its importance in digital circuitry.

An operational amplifier has a very high dc gain (open loop), flat response over a wide frequency range, very high input impedance, and very low output impedance. In Fig. 7-3, one input terminal has a minus sign and the other has a plus sign. The minus input terminal is an inverting input; this means that the signal on this terminal is inverted in polarity at the output. The plus input terminal is a noninverting input; this means that the signal on this input terminal has the same polarity at the output.

If the same signal is applied to both inputs, the output voltage is zero. This is called a differential input, since the output is actually proportional to E2 − E1.

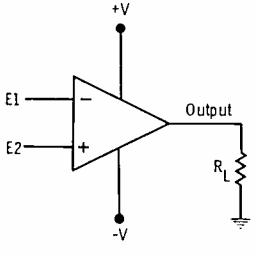


Fig. 7-3. Basic op-amp for voltage comparator.

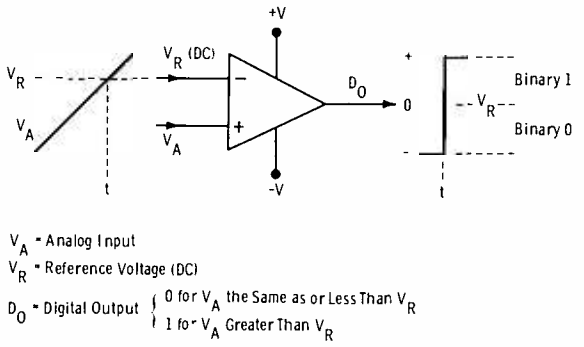


Fig. 7-4. Basic operation of voltage comparator.

In Fig. 7-4, assume that an analog signal (V_A) is applied to the noninverting input, and some reference dc voltage (V_R) is applied to the inverting input. When the two voltages are exactly equal, zero output occurs. This reference output and all values below it represent binary 0. If the analog voltage exceeds V_R, the open-loop op amp rapidly saturates to +V, producing a binary 1. This indicates that the instantaneous analog voltage *exceeds* the reference voltage.

7-4. BASIC CODEC WITH COMPARATOR

Some codecs that are used as indicating devices employ ramp-comparison converters for adc, with readout decoders. The ramp-comparison method of converting analog information to digital information is illustrated basically in Fig. 7-5. The familiar sine wave is shown to represent an analog voltage. The sine wave may be seen to be analogous to rotation of a potentiometer between plus and minus values through a zero reference voltage. (The input source could just as easily be the secondary of a transformer with the primary connected to a microphone, phono pickup, synchro system, etc.) The potentiometer provides one input to a voltage-comparator circuit. The position of the potentiometer at any instant is termed the *analog address.*

The other input to the voltage comparator is a linear ramp voltage from the function generator. The ramp is initiated from a start signal, which also enables a gate circuit to allow continuous-running clock pulses to pass to the digital counter (flip-flops). As long as there is a difference in magnitude between the analog and function-generator inputs to the comparator, clock pulses at a constant repetition rate are permitted to reach the counter. When the linearly rising ramp voltage reaches equality with the analog address, a stop signal is generated by the comparator; this signal disables the gate

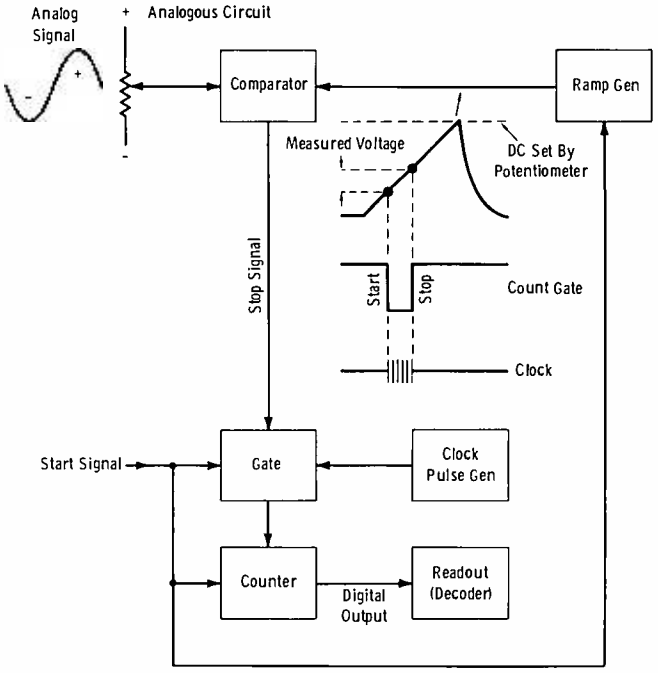


Fig. 7-5. One type of codec with comparator and ramp generator.

and blocks the clock pulses from the counter. The number of pulses accumulated in the counter during the comparison time interval is proportional to the amplitude of the analog input voltage at the time of address, and is the desired digital information.

In this example, the readout device with its associated decoder serves as the end result. It is not, in the strict sense, a dac, since the output is a series of numbers representing the original analog value. However, the same arrangement can be used to trace curves, to operate columns of LEDs as audio peak indicators, or otherwise represent the analog signal. Varieties of dac's are covered later in this chapter. Just bear in mind that a decoder need not necessarily be a dac.

7-5. ADC BY DIRECT COMPARISON

Fig. 7-6 shows a separate voltage comparator for each possible voltage increment. In this example, each voltage increment (quantum) is 1 volt. For n bits of binary information, the system requires $2^n - 1$ comparators (no comparator is required for zero). In the

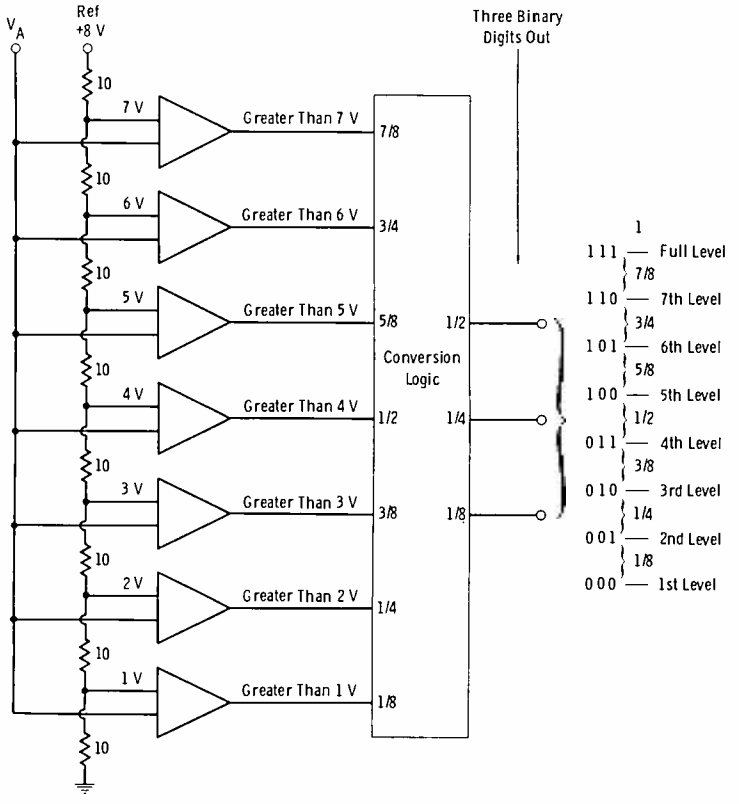


Fig. 7-6. Direct comparison, simultaneous conversion method of adc.

example of Fig. 7-6, there are eight possible levels requiring $8 - 1 = 7$ comparators. The output is a 3-bit binary word ($2^3 = 8$). If eight bits were used, there would be $2^8 = 256$ possible levels, requiring 255 comparators. Note that this would be far more accurate than the 3-bit output of Fig. 7-6.

The output of the parallel comparators is not directly usable information. These outputs must be converted to n bits of binary information in some binary-coded form. In the example of Fig. 7-6, each voltage level is converted to a 3-bit binary word that describes one of eight possible voltage levels.

In this method, all bits of the digital representation are determined simultaneously due to the parallel configuration. This is sometimes called *flash encoding* and is one of the fastest encoding methods in existence. Its main disadvantage is that for large values of n, the massiveness of large numbers of comparators and the associated

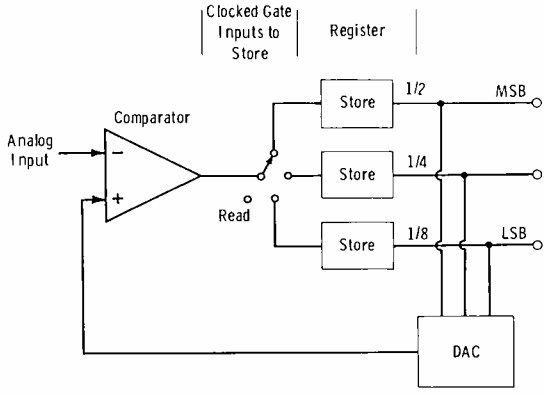


Fig. 7-7. Simplified three-bit successive-approximation adc.

conversion logic may be prohibitive. However, this method has been used for digitizing video signals.

7-6. SUCCESSIVE APPROXIMATION

The successive-approximation adc employs not only a comparator, but also a digital-to-analog converter (dac) as an approximation feedback to the comparator. It is an excellent compromise between circuit complexity, speed, and ability to produce high-accuracy codes.

Fig. 7-7 illustrates a simplified version of the successive-approximation method. Note that the analog signal is applied to one input of the comparator. The other input is from the dac. In this technique, the digital output is determined one bit at a time, starting with the most significant bit (MSB). Note from Table 7-1 that the MSB of any binary word first occurs at half level. At the start (see Fig. 7-8), the varying analog input is first compared with one-half of the full-scale voltage by setting the MSB in the register to 1. If the analog input is greater than this first approximation, the second most significant bit is also set to 1. This causes the input to be compared with three-fourths of the full-scale voltage. Conversely, if the analog input is less than the first approximation, the first register bit is reset to 0 while the second is set to 1, causing the input to be compared with one-fourth of the full-scale voltage. This is made clear by following Fig. 7-8 between clock pulses 1 and 2.

Note from Fig. 7-8 that, in like manner, the analog input is compared with successively finer approximations until the LSB has been determined. In this example (three bits), each increment is one-eighth of the full-scale voltage, or one part in eight. A 5-bit binary

Table 7-1. Demonstration That MSB First Occurs at Half-Level

	Binary Number	Decimal Number	
	1111	15	← 16th Level
	1110	14	
	1101	13	
	1100	12	
	1011	11	
	1010	10	
MSB First Occurs for 4 Bits →	1001	9	
	1000	8	← Half-Level for 4 Bits
	0111	7	
	0110	6	
MSB First Occurs for 3 Bits →	0101	5	
	0100	4	← Half-Level for 3 Bits
MSB First Occurs for 2 Bits →	0011	3	
	0010	2	← Half-Level for 2 Bits
	0001	1	
	0000	0	← First Level

word would have increments of 1/32, or one part in 32. An 8-bit word would have increments of 1/256, or one part in 256.

Details of the successive-approximation adc along with associated "sample-and-hold" circuitry are covered under pulse code modulation (pcm) in Chapter 8.

7-7. MULTIPLE-COMPARISON SUBRANGING ADC

In the simultaneous method of adc (Section 7-5), only one "look" is required for conversion of any number of bits. In the successive-approximation technique of Section 7-6, note that three "looks" are required for three bits. An 8-bit binary word would require eight looks. Thus, the latter method is slower than the simultaneous method, but it employs much less hardware.

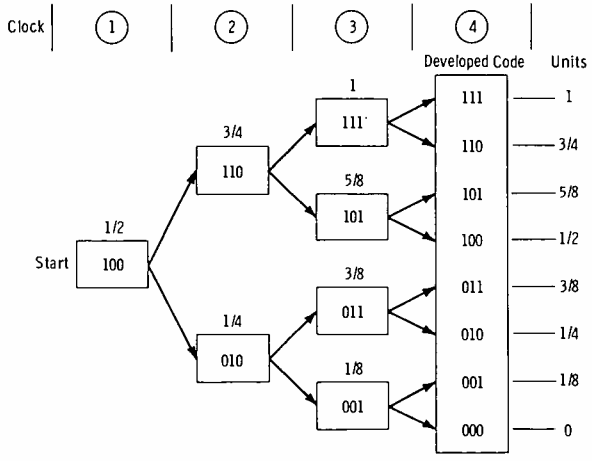


Fig. 7-8. Three-bit successive approximation develops 0-1 in increments of 1/8.

The multiple-comparison subranging technique (Fig. 7-9) is a compromise between the two methods just described. The system in Fig. 7-9 has four subranges per step; hence, only two looks are required for an 8-bit binary. This method is therefore much faster than the successive-approximation method, yet it is only slightly slower than the simultaneous conversion method requiring 255 comparators. Only 30 comparators are necessary for eight bits, as indicated in Fig. 7-9.

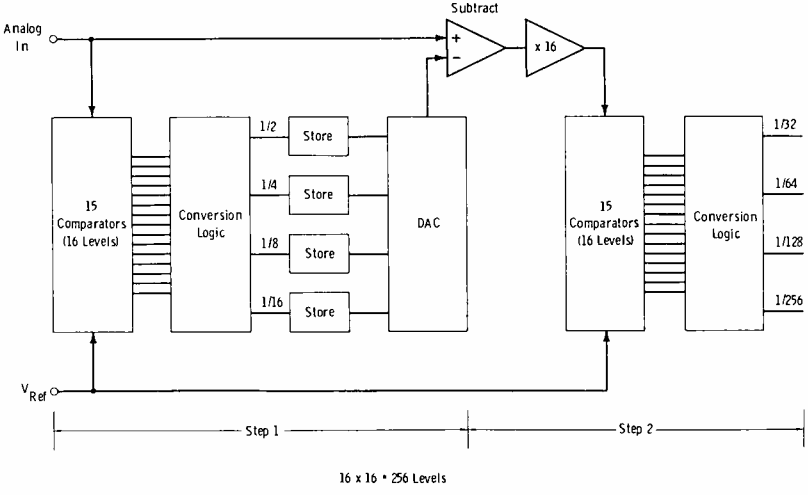


Fig. 7-9. Example of eight-bit multiple-comparison subranging adc.

The first sixteen-level, one-look (flash-encoded) adc yields four bits that are placed in temporary storage. These will become the most significant four bits of the final binary word. In the second and final step, the four bits drive a dac whose output is subtracted from the input to obtain a difference. The resultant is amplified 16 times, and the process of Step 1 is repeated to obtain the least significant four bits. This subrange can be divided into four more subranges, and the same process repeated for binary words that are longer than eight bits.

7-8. MECHANICAL ADC

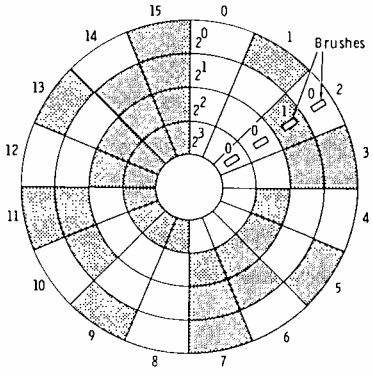
An *adc* is defined as a device that receives an analog input and supplies a digital output. A common mechanical application is shaft position (analog) that can be expressed in digital form. This technique is sometimes necessary in synchronizing a number of recording devices that employ rotating capstans driven by a motor shaft, for servos, and for other mechanical devices.

The coded disc (Fig. 7-10A) is one device used for adapting a binary code to describe shaft position. In this drawing, the shaded areas represent conductors, the light areas represent insulators, and the small rectangles represent brushes. In optoelectronic pickups, the shaded areas would represent solid background (opaque), the light areas transparent sections, and the small rectangles the optoelectronic pickups. Magnetic pickups may also be used.

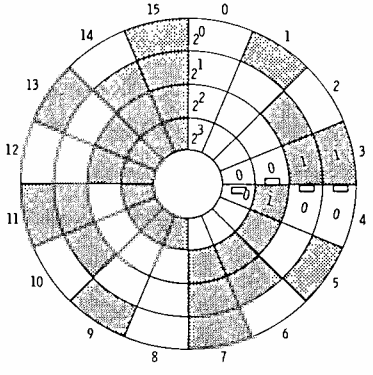
The disc is attached to the associated shaft so that as the disc rotates in a counterclockwise direction under the brushes, an "on" (1) signal is generated each time a brush contacts a dark area, and an "off" (0) signal is generated each time a brush contacts a light (unshaded) area. The circles that produce values of 2^0, 2^1, 2^2, and 2^3 are labeled at the 0 position (segment) of the wheel. Thus, the outermost circle produces the least significant digit (LSD) in the code, followed by increasing orders to the most significant digit (MSD) produced by the innermost circle.

The binary number represented by each segment of the disc is read by interpreting the dark and light areas in order from the innermost to the outermost circle. Read a "0" for each light area and a "1" for each dark area. The areas in segment 1 are read 0001, representing the number 1 in the decimal system. The areas in segment 12 are read 1100, representing the number 12 in the decimal system. In Fig. 7-10A, the brushes are shown in a position that reads 0010, or decimal 2.

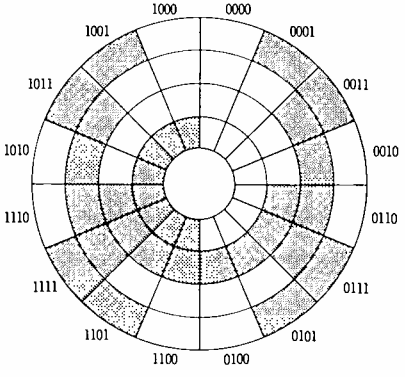
Because only four zones are used (2^0, 2^1, 2^2, and 2^3), the resolution of the disc is one part in 16 (counting zero as one part). If an additional circle were added, a resolution of one part in 32 could



(*A*) *Binary coded disc.*



(*B*) *Misaligned brushes.*



(*C*) *Disc with Gray code.*

Fig. 7-10. Application of coded disc for rotating shaft.

be obtained. The resolution can thus be increased by a factor of two for each circle that is added.

A circuit can be completed through the disc by connecting each conducting segment to a conductor on the rear of the disc. This common conductor then would serve as a slip ring.

The major difficulty with the coded disc described above is illustrated by considering the disc to be in a position where the output number is changing from 0011 to 0100 (Fig. 7-10B). Because the brushes used for readout from the disc cannot be perfectly aligned, it is probable that all bits will not change simultaneously. If the shaft stops at a position where the 2^2 brush is in the conductive (shaded) area and the 2^1 and 2^0 brushes are not completely in the insulated (unshaded) areas, the output will be 0111. This represents an appreciable error. If, on the other hand, the 2^2 brush is not in the shaded area and the 2^1 and 2^0 brushes are in the unshaded areas, the output will read 0000. This again represents a large error.

The possibility of large error is most severe when more than one bit changes value simultaneously. Table 2-2 (Chapter 2) reveals how frequently this condition arises. Starting with 0000 in the natural binary code or bcd code, alternate numbers require a change in more than one bit. Thus, the possibility of error is great, and this code is not always satisfactory for use in shaft-position digitation.

The most popular code used to circumvent this problem is the *Gray code* (Fig. 7-10C), sometimes called the *unit distance* code. The Gray Code representation of numbers from 0 to 15 is shown. Note that only one bit changes value from one position to the next higher or lower position. In this way, the magnitude of the error introduced by imperfect alignment of the read-out brushes or by reading out data from a brush (or brushes) that spans a shaded and unshaded section cannot exceed 1. Thus, a 4 may be read for a 5, or a 7 may be read for an 8, but larger errors cannot be made .

The same type of errors can be made with optoelectronic devices if the light sources and pickups are not in perfect alignment across the disc.

7-9. DAC PRINCIPLES

In our analog world, data normally acquired from anything that is tested or measured are in analog form that is difficult to handle, process, or store for later use without introducing considerable error. Therefore, if data are subject to much handling after they are acquired, they are safely digitized with little chance of error accumulation in successive manipulation. Digital data are readily presented in numerical form regardless of the number of bits, and are just as easily manipulated, processed, and stored. Digitized data can be

processed mathematically, sorted, analyzed, and used for control much more accurately and rapidly than can analog data. However, once this digital storage and necessary manipulation are accomplished, the digital-to-analog converter (dac) is often required to interface the digitized data with the analog world.

A dac is based primarily on *ladder network* theory. There are two basic types of ladder networks:

1. Binary-weighted resistor ladder network.
2. R-2R ladder network.

Binary-Weighted Resistor Ladder

The most straightforward ladder network consists of resistors whose values are weighted according to the 1-2-4-8 . . . n binary system.

Fig. 7-11 illustrates how a linear summing amplifier and a binary-weighted resistor ladder are employed for digital-to-analog conversion. The logic switches (such as could be operated from transistor switches) determine whether +5 volts or ground is applied to each of the various inputs. In this example, input 1 (I1) is the MSB, and I6 is the LSB since the associated resistor is 32 times the value of R (6-bit binary). In Fig. 7-11, the logic switches are shown in the positions that represent binary 110010.

If all inputs are grounded, minimum output level occurs. A single switch in the 1 position produces a corresponding increase in output depending on the weighted value of the associated resistor. If all switches are in the 1 position, maximum output results. Thus, the output is an analog representation of the digital input code.

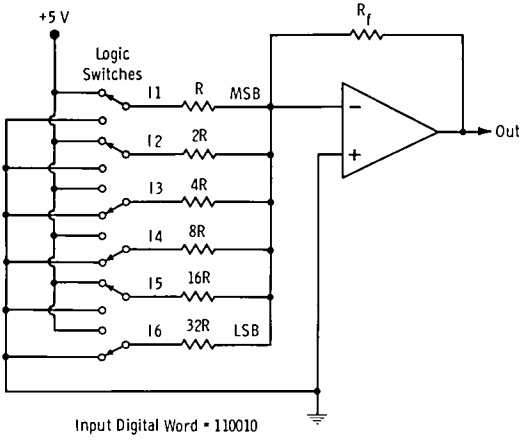


Fig. 7-11. Summing amplifier for dac using binary-weighted ladder network.

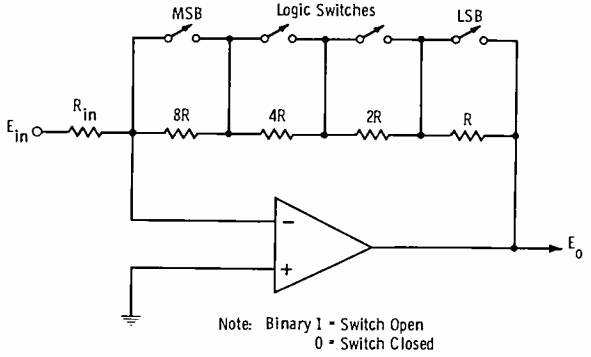


Fig. 7-12. Four-bit dac using operational amplifier with variable feedback.

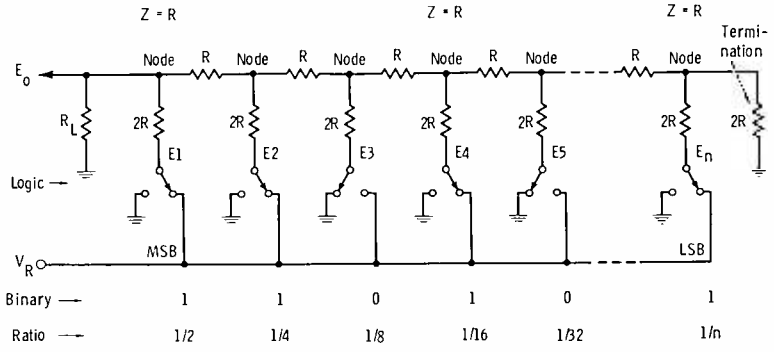
The operational amplifier of Fig. 7-12 has a feedback loop with a series array of binary-weighted resistors. Each resistor is bypassed with a switch which is either open or closed. For a binary 0, the switch is closed. For a binary 1, the switch is open. If all contacts are closed (0000), the feedback path has zero resistance (a direct short between output and input), and no voltage gain is produced. This is equivalent to a voltage follower. When a binary 1 input is applied, the switch opens, and resistance is inserted in the feedback path. This produces a voltage gain that is equivalent to the ratio of the feedback resistance to R_{in}. Therefore, E_o represents the analog value of a digital function. In common with all dac's, the greater the number of switches (the greater the number of binary digits), the smaller the possible incremental difference in output voltage levels, and the higher the *resolution.*

The major point of concern with the binary-weighted ladder network is accuracy of the resistors. In order for the dac to be able to detect the difference of one LSB change at the input, the resistors must have a very tight tolerance. A tolerance of 0.1% or better is generally required. This becomes very difficult with binary words having a large number of bits. Note that for an 8-bit binary, the LSB would be 128R. Thus, if R (MSB) were 10K, the LSB resistor would be 1.28 megohms. Resistor accuracy and temperature coefficients in large binary-weight resistor networks can be troublesome and costly.

The R-2R Resistor Ladder Network

The R-2R ladder network was developed to overcome the magnitude and tolerance requirements of large binary-weighted networks. The R-2R technique is widely used in system designs of recent years.

The R-2R network has an output impedance that is constant regardless of the number of stages, maintains the binary weighting of



Note: V_R = Constant-Voltage Reference Source (Very Low Internal Resistance)

Fig. 7-13. Example of R-2R dac voltage ladder network.

the input signals, but uses many equal-value resistors rather than different-value resistors. In practice, only the values R and 2R are required.

The R-2R ladder network may be used with constant-voltage or constant-current sources. The constant-voltage-source application is shown in Fig. 7-13. Note that this type of ladder network has the appearance of a discrete representation of a transmission line. If there were an unlimited (infinite) number of stages, then the impedance at the left side of any node looking to the right would be a constant value (R).

To achieve this constant-impedance characteristic for a finite-length ladder network, the ladder is terminated in its characteristic impedance (the 2R resistor to ground at the far right of Fig. 7-13). This eliminates impedance mismatching. The impedance looking to the right from the right side of any node is 2R, which means that a value of 2R is required as terminating resistor. Note that for a simplified drawing, the LSB resistor in the network is, in effect, the impedance of R.

The logic switches in Fig. 7-13 are shown in the positions for binary 110101. The 1 or 0 is obtained by switching between ground and the constant-voltage source. Remember that a constant-voltage source has extremely low internal impedance (ideally zero), and the constant impedance of the network is not disturbed by 1 or 0 binary selection. In essence, each leg of the network provides a voltage division by two. For six bits, the output for the MSB is 32 times the output for the LSB. This maintains the binary weighting of the input.

Fig. 7-14 illustrates an R-2R network with a constant-current source for a 3-bit binary. A zero current is delivered to the network

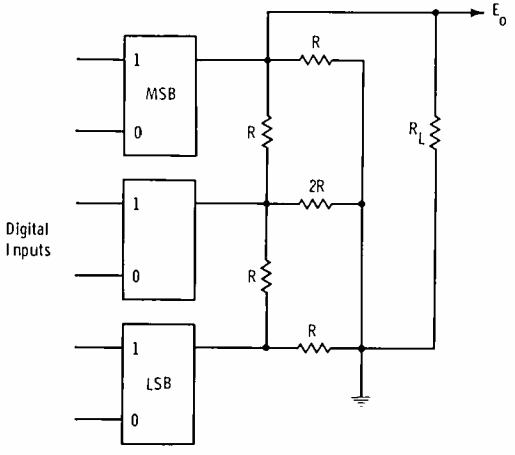


Fig. 7-14. An R-2R dac network with three-bit constant-current source.

for each 0 bit, and a constant current, I, is delivered for each 1 bit. The voltage output (E_o) for this network is:

$$E_o = \frac{IR}{3}\left(\frac{R_L}{R_L + \frac{2}{3}R}\right)\frac{N}{2^{n-2}}$$

where,

n is the number of bits in a given word,
N is the magnitude of the digital number (base-10 value).

This is a current-summation circuit and is somewhat more commonly encountered than the R-2R voltage application.

To see how the R-2R current network of Fig. 7-14 "tracks" between decimal values 1 and 7, first assume the following:

$$\begin{aligned} R &= 1000 \text{ ohms} \\ R_L &= 1000 \text{ ohms} \\ N &= 7 \text{ (binary 111)} \\ n &= 3 \text{ (number of bits used)} \end{aligned}$$

For decimal 7:

$$\begin{aligned} E_o &= \frac{I(1000)}{3}\left(\frac{1000}{1000 + (2)(1000)/3}\right)\frac{7}{2^{3-2}} \\ &= \frac{I(1000)}{3}\left(\frac{1000}{1666}\right)\frac{7}{2} \\ &= 333I\,(0.6)(3.5) \\ &= 333I\,(2.1) \\ &= 700I \end{aligned}$$

Now assume that the flip-flops produce a current (I) of 10 mA:

$$E_o = 700(0.01) = 7 \text{ volts}$$

Thus, E_o for decimal 7 (binary 111) is 7 volts. Now find the output of the same circuit for decimal 1 (binary 001). Since the only change is N = 1:

$$E_o = 333I\,(0.6)\frac{1}{2}$$
$$= 333I\,(0.3)$$
$$= 100I$$

Then with I = 10 mA:

$$E_o = 100(0.01) = 1 \text{ volt}$$

The voltage ratio is 7 volts to 1 volt. Thus, the required voltage ratio exists for decimal 7 and 1.

EXERCISES

Q7-1. Define (A) adc, (B) dac, (C) codec.

Q7-2. If the peak-to-peak signal voltage is 1 volt, what is the voltage value of one quantum in an 8-bit binary codec?

Q7-3. An adc using direct comparators for each voltage level requires how many comparators for a 10-bit binary?

Q7-4. What is the main advantage of a multiple-comparison subranging adc over other methods?

Q7-5. (A) Define the unit-distance code.
(B) For what application is it most suitable?

Q7-6. For the circuit of Fig. 7-14, find E_o for the following conditions:
(A) R = 500 ohms
R_L = 1000 ohms
N = 6 (binary 110)
n = 3
Flip-flop output = 5 mA
(B) Same as in (A) except N = 3 (binary 011).

CHAPTER 8

Data Links

A data link consists of the various electronic devices necessary to allow automatic transmission of information from one point to another. The ability to transmit *automatically* is the function that distinguishes data links from other forms of communication. The data link is composed of *hardware* (the physical equipment forming the digital system) and *software* (the programming format and all other techniques not included in the term "hardware"). Common applications of data links in broadcasting are:

1. Transmitter remote control
2. Automatic logging devices
3. Computer-controlled switching systems

Data is a term used to designate the electrical signals that represent information. Information can take many forms: words, numbers, symbols, audio, video, etc. The encode-transmit section accepts the information in whatever form it can handle and encodes it into a form suitable for transmission over the system path. The path may be through wires or cable or over the air. The decode-receive section must handle this data signal plus noise. Errors that may result from noise or other causes must be detected and recognized. If an error occurs, the signal must be retransmitted or otherwise corrected. The decoded output is the original information content of the data signal.

Information flow can be *on-line*, which permits two or more digital units to communicate directly, or the output of one or more digital units can be fed into a storage device to establish communication *off-line*. Data links can be basically classified as either baseband, modulated baseband, or modulated carrier. Data communication channels can be further classified as simplex, half-duplex, or

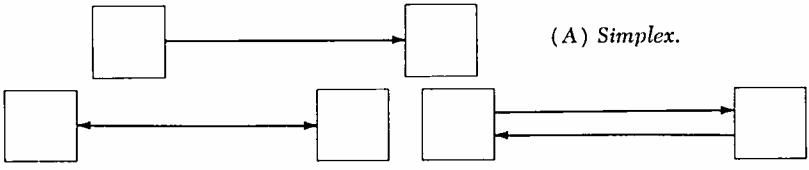


Fig. 8-1. Basic data-transmission methods.

full duplex, as shown in Fig. 8-1. This classification specifies whether the channel can transmit data in one direction only (Fig. 8-1A), one direction at a time (Fig. 8-1B), or both directions simultaneously (Fig. 8-1C). A two-wire circuit can be used for full duplex by employing frequency-division multiplexing.

8-1. BASEBAND SYSTEMS

A baseband system transmits the coded data signals directly on the transmission path without modulation. Pulse formats of baseband signals have been covered previously in Sections 1-3 and 6-3. Fig. 1-2 (Chapter 1) shows the fundamental baseband and modulated pulse formats. In practice, *bipolar* pulses are also encountered, as illustrated in Fig. 8-2. Waveforms B through D are bipolar pulses equivalent to the conventional RZ pulse train (Fig. 8-2A) that represents binary 1011001. Fig. 8-2 may be analyzed as follows:

A. Conventional RZ baseband signal. Review Section 1-3.

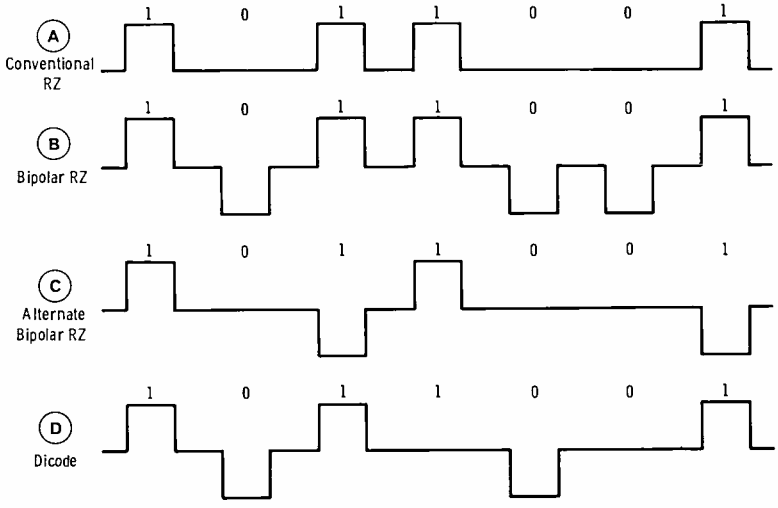


Fig. 8-2. Typical bipolar coding techniques.

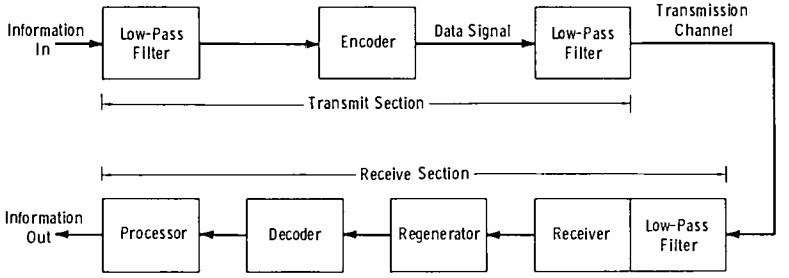


Fig. 8-3. Simplified block diagram of baseband transmission.

B. Bipolar RZ. Short positive current for 1 bits, negative for 0 bits. Return to zero current for every bit. Always two transitions made in each unit of time. Requires no sampling since there is a pulse for every bit.
C. Alternate bipolar RZ. Short pulses in alternating directions for successive 1 bits. Immediate return to zero current. No change for 0 bits. Requires sampling to determine state of each bit.
D. Dicode. Bipolar bit occurs on each new series of pulses. Each time the bit changes from 1 to 0 or from 0 to 1, a pulse (either negative for 0 or positive for 1) is sent. Requires sampling to determine state of each bit.

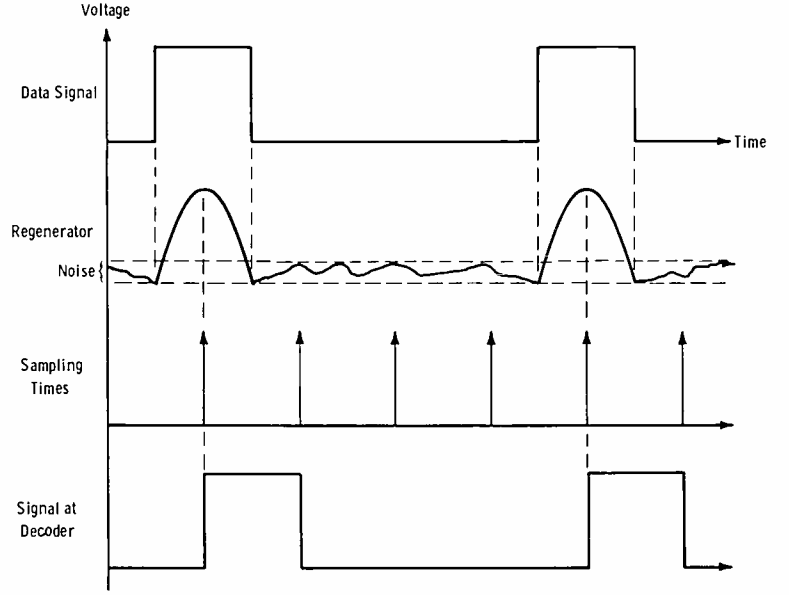


Fig. 8-4. Typical baseband-system waveforms.

There are many more types of coding used than those outlined in this book. Space permits covering only those most frequently encountered in broadcast and general communications systems.

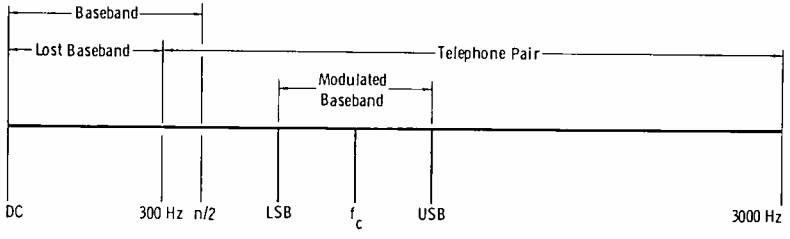
Baseband transmission is generally used only between points at a given plant or location, not for long-distance transmission. Fig. 8-3 illustrates a simplified baseband application. Fast-rise-time pulses require an unlimited frequency band; hence, no attempt is made to preserve the rectangular shape of pulses formed in encoding digital data.

As shown in Fig. 8-3, a low-pass filter is used at the input to the transmit section to eliminate all frequencies above the passband of the particular transmission path. The same type of filter may be used at the feed point to remove encoding frequency transients. The low-pass filter at the receive point eliminates any transient pickup (noise) on the path, and a regenerator circuit is used to reconstruct the data signal.

Fig. 8-4 is a plot of the original data signal from the transmit encoder, the received signal (low-pass filtered), and the regenerated wave. The regenerator samples each incident pulse at its midpoint for maximum reliability, which causes the regenerated data signal to be advanced in time one-half of the pulse duration.

8-2. THE NEED FOR MODULATION

A baseband signal transmitted at a steady rate of n bits per second may have a bandwidth ranging from 0 Hz (dc) when several 1 or 0 bits occur in series, to n/2 Hz when a series of alternate 1 and 0 bits (the highest frequency) is sent. If these data signals are to be sent over a relatively long path such as on a telephone pair, the dc and low-frequency ac components are quickly absorbed by the internal resistance, and only noise remains. The typical response of an unequalized telephone pair is 300 to 3000 Hz. If trans-



f_c = Carrier Frequency
LSB = Lower Sideband
USB = Upper Sideband

Fig. 8-5. Relocation of baseband signal through modulation.

mission of a baseband signal is attempted, the transmission system will exhibit a cutoff somewhere between 0 and n/2 Hz (see Fig. 8-5). However, if this baseband signal is used to modulate a carrier within the voice band, the entire bandwith from 0 to n/2 (the carrier frequency can represent the dc component) will be relocated within the 300–3000 Hz passband, as shown in Fig. 8-5.

The use of modulation can also result in more efficient use of available bandwidth. For example, a baseband teletypewriter signal has a bandwidth of up to 150 Hz, which is a very small portion of the voice-band range of 300 to 3000 Hz. This is an actual frequency range of $3000 - 300 = 2700$ Hz. About 15 teletypewriter channels, with adequate guard bands between frequencies, can be carried on one pair of wires when modulation is used. This capacity can practically be doubled through the use of single-sideband transmission.

8-3. THE SAMPLING PROCESS

The basic sampling process was described in Section 7-1 of the previous chapter. Sampling is, in effect, an impulse modulator. If the input analog signal of Fig. 8-6A is sampled with impulses of unit amplitude occurring once every t seconds (Fig. 8-6B), the result is a series of impulses at t intervals whose amplitude varies as shown in Fig. 8-6C. This signal is termed *pulse-amplitude modulation (pam).*

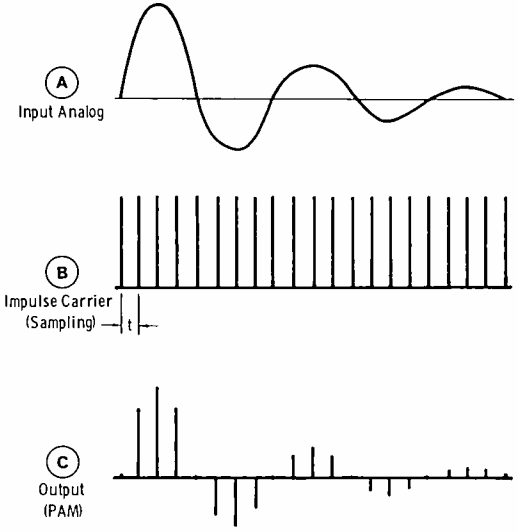


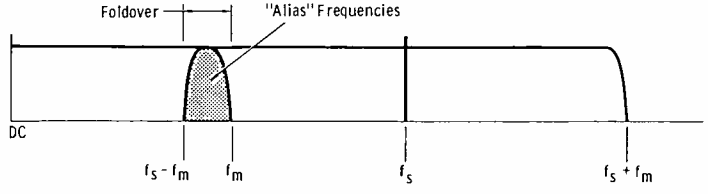
Fig. 8-6. Sampling as impulse modulation.

The Sampling Theorem

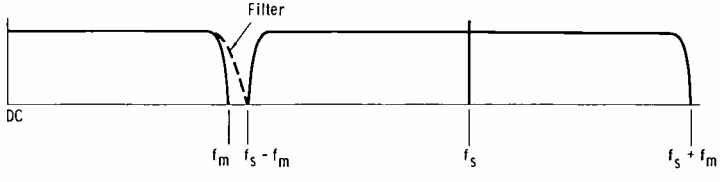
The sampling theorem states that if a time-varying signal whose bandwidth is B hertz is instantaneously sampled at regular intervals at a rate slightly greater than 2B samples per second, the resulting samples contain all the information of the original signal. The minimum sampling rate of 2B pulses per second is known as the *Nyquist rate.* The maximum period between sampling pulses is 1/2B seconds and is known as the *Nyquist period.* Proof of the sampling theorem is very involved mathematically and will not be given here.

For all intents and purposes, the sampling frequency (f_s) may be considered a carrier. When a signal of frequency f_m is sampled by f_s, sidebands at $f_s - f_m$ and $f_s + f_m$ occur. Fig. 8-7A shows what happens if the sampling frequency f_s is less than twice the frequency f_m. The lower sideband ($f_s - f_m$) overlaps the upper region of the modulating signal, resulting in image frequencies and beat frequencies that cannot be separated by filtering action. Conversely, if the sampling frequency is greater than twice the highest modulating frequency (Fig. 8-7B), overlap does not occur and efficient filtering action is possible.

The importance of proper filtering action (Fig. 8-3) can now be observed in light of the sampling theorem. The information input must be filtered to prevent frequencies higher than half the sampling frequency from entering the encoder where sampling takes place. Another low-pass filter at the encoder output (line feed) prevents harmonics of f_s from being transmitted. A low-pass filter



(A) Sampling frequency less than $2f_m$.



(B) Sampling frequency greater than $2f_m$.

Fig. 8-7. Spectrum of sampling frequency.

at the receiver prevents high-frequency noise and impulse transients from entering the regenerator.

Any baseband signal that is sampled is considered to be a modulated baseband signal. For the previously mentioned teletypewriter signal of 150-Hz bandwith, the sampling frequency would need to be at least 300 Hz. To place the lower sideband above the 300-Hz cutoff region of the telephone line, the sampling frequency would need to be $150 + 300 = 450$ Hz. Other teletypewriter signals to be transmitted (multiplexed) on the same telephone pair could have higher sampling rates, and bandpass filters could be employed at both ends of the path to prevent interference. Or, the signals could be used to modulate higher-frequency carriers, with suitable bandpass filters at the sending and receiving points to separate the individual signals. Multiplexing techniques are discussed later in this chapter.

Sampling Theorem for Modulated Carrier

When the information bandwidth (B) is displaced from zero, it extends from some frequency f1 to a higher frequency f2 so that $f2 - f1 = B$. A question that often arises in the student's mind at this point is, "What happens to the minimum sampling-frequency requirement? Is it still 2B, or is it 2f2?" This is a valid question, and the solution points to an important relationship between choice of carrier frequency (when required) and the proper sampling rate.

When a band of frequencies from f1 to f2, displaced from zero, is sampled, the minimum sampling rate is:

$$f_s = 2B(1 + k/m)$$

where,

f_s is the minimum sampling frequency,
m is the largest integer (whole number) not exceeding f2/B,
k is $f2/B - m$,
B is the bandwidth $= f2 - 1$.

For example, find the minimum sampling rate for an information channel extending from 135 kHz to 150 kHz. First, it is realized that this signal could be sampled at 2×150 kHz $= 300$ kHz; also, the frequency band could be translated down 135 kHz, so that it extended from 0 to 15 kHz, and then sampled at 30 kHz. But, the latter procedure would needlessly require additional equipment.

In this example, f1 = 135 kHz and f2 = 150 kHz. Therefore, B = 15 kHz. To apply the formula for f_s, it is necessary first to find m and k:

$$m = \frac{f2}{B} = \frac{150}{15} = 10$$

$$k = \frac{f2}{B} - m = \frac{150}{15} - 10 = 0$$

Substituting these values in the formula gives:

$$f_s = 2(15)(1 + 0/10) = 30\ \text{kHz}$$

Note that this is the same minimum sampling rate required for a 0–15 kHz bandwidth. Thus, no increase in sampling rate was required in this example of going to a higher frequency range. However, this is not always true. For example, let the bandwidth remain the same, but increase the frequency range of interest to f1 = 145 kHz, f2 = 160 kHz. Now:

$$m = \frac{160}{15} = 10.66 = 10$$

$$k = \frac{160}{15} - 10 = 0.66$$

$$f_s = 2(15)(1 + 0.66/10)$$
$$= 30(1.066) = 32\ \text{kHz}$$

Note that the minimum sampling rate is higher than that of the first example by 2 kHz.

Now assume that the frequency range is further increased so that f1 = 150 kHz and f2 = 165 kHz. This gives m = 11 and k = 0. The minimum sampling frequency is $f_s = 2 \times 15\ \text{kHz} = 30\ \text{kHz}$, as in the first example.

This emphasizes that a proper choice of carrier frequency relative to signal bandwidth requires a lower sampling frequency that need not exceed the Nyquist rate for the baseband signal. Note that the value of k can vary between zero and unity. When the band is located between adjacent multiples of B, k = 0, and the minimum sampling rate is 2B no matter how high the frequency range of the signal may be. The sampling rate is a minimum when k = 0 and increases from 2B to a maximum of 2B(1 +1/m) as k approaches unity. This relationship is valid because the value of m is given as the largest integer not exceeding f2/B; this makes k the difference between f2/B and the largest integer.

It should also be pointed out that the minimum sampling rate in practice is usually somewhat greater than twice the highest frequency in the signal (by about 15%), to compensate for accumulated effects in a sampling-type digital system.

8-4. PULSE-AMPLITUDE MODULATION (PAM)

In *pulse-amplitude modulation*, a pulse with an amplitude proportional to the data or analog signal amplitude at the moment of sam-

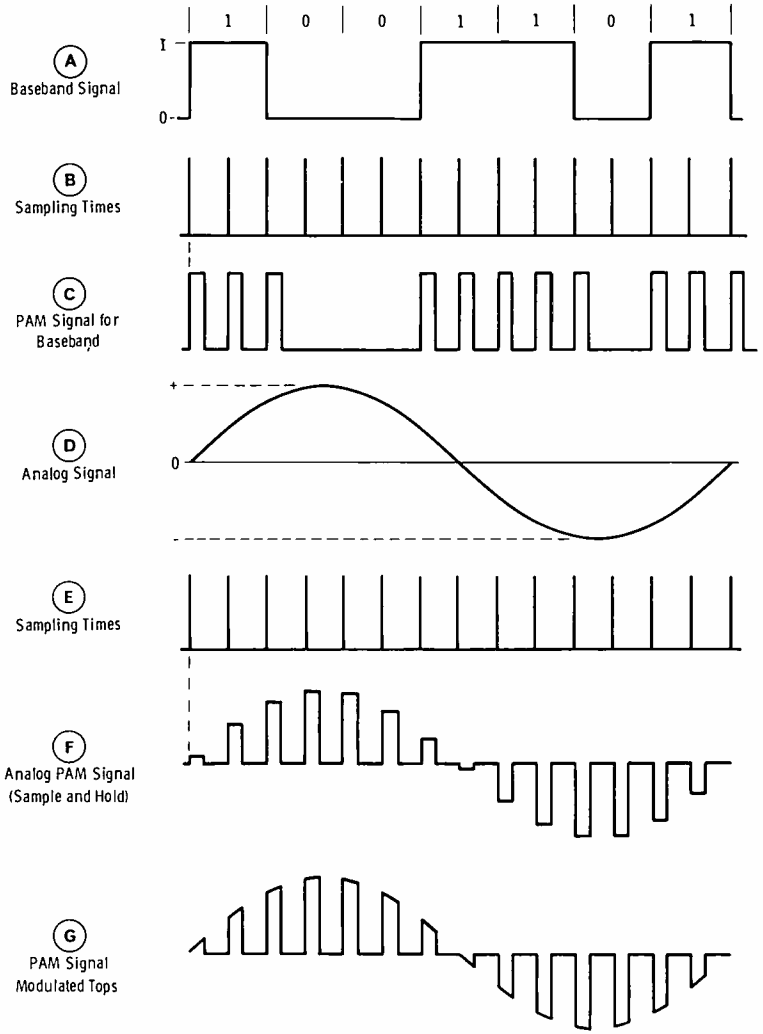


Fig. 8-8. Pulse-amplitude modulation (pam).

pling is generated. Because of the constant sampling frequency, the pulses are evenly spaced. Fig. 8-8 illustrates the pam function; it may be analyzed as follows:

A. Baseband signal representing binary 1001101. A baseband signal is of single polarity—either essentially zero to unity in the positive direction, or a negative potential to a less negative potential. In either case, the signal never reverses polarity.

B. Represents sampling times of the baseband signal.
C. Resulting pam signal for the baseband signal of (A) never reverses polarity.
D. Analog signal representing ac with changing polarity.
E. Sampling times for signal of (D).
F. Pam for analog signal of double polarity. Note that in this example, the leading edge of the pulse is made at the sampling time, and by means of a "sample-and-hold" circuit, the tops of the pulses are flat.
G. When a sample-and-hold circuit is not employed, the tops of the pulses follow the analog signal for the duration of each pulse.

From Fig. 8-7, it may be noted that the sampling frequency generates sidebands equivalent to those of a dsb amplitude-modulated carrier. In single-polarity pam, harmonics of the sampling frequency and sidebands exist far beyond the limits shown in Fig. 8-7. For double-polarity pam, the carrier (f_s) and its harmonics are missing, leaving only the upper and lower sidebands and the modulation frequency itself. This is similar to double-sideband suppressed-carrier modulation.

You may wonder why a baseband signal is ever pulse-modulated. The most important reason is to overcome noise. A pulse-modulation system uses a carrier of short duration and high peak power, which has advantages in obtaining a more reliable signal-to-noise ratio.

The demodulation process for pulse-modulated signals is very simple. Since the modulating signal itself is present, just insert a low-pass filter designed to pass the modulating signal but to reject the carrier frequencies and their respective sidebands. Generation of pam signals is discussed under pcm in this chapter.

8-5. PULSE-DURATION MODULATION (PDM)

For *pulse-duration modulation* (also termed *pulse-width modulation*, or *pwm*), the duration of the generated pulses is proportional to the amplitude of the data signal, as shown in Fig. 8-9A. For this baseband signal, an arbitrary pulse duration is assigned to represent a space (binary 0) at the sampling instant, and a greater pulse duration represents the mark (binary 1). Thus, only two specified durations occur; one for 0 and another for 1.

For the analog signal of Fig. 8-9B, the pdm signal varies linearly between two extremes which represent the maximum excursions of the analog signal. In this example, the widest pulses represent maximum positive excursions, and the narrowest pulses represent maximum negative excursions. Zero-axis crossing is an arbitrarily

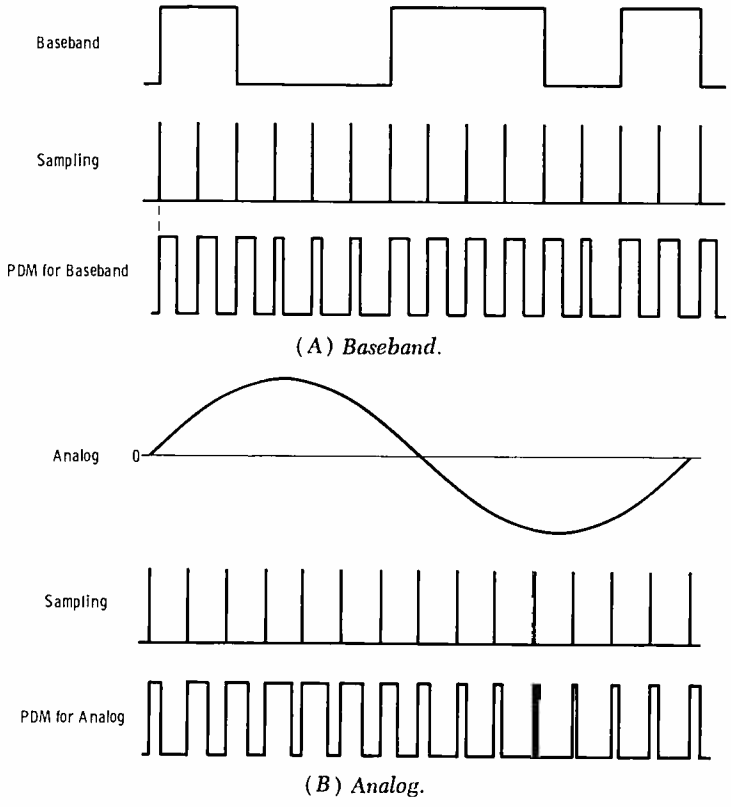


Fig. 8-9. Pulse-duration modulation (pdm).

fixed pulse duration. Thus, the pdm signal is a single-polarity signal for a double-polarity analog input.

In Fig. 8-9, the leading edge of the pdm pulse is always initiated by the sample time, and the time of occurrence of the trailing edge is varied by the amplitude of the data or analog signal. It is also possible for pdm to be accomplished by varying the occurrence of the leading edge.

8-6. PULSE-POSITION MODULATION (PPM)

In the process of *pulse-position modulation (ppm),* the exact time of sending the pulse varies within the constant sampling period according to the amplitude of the data signal, as shown in Fig. 8-10. This figure is an example in which pdm is used as an intermediate process to derive a ppm signal. The pdm signal of Fig. 8-10A represents the instantaneous amplitudes of the original sampled signal.

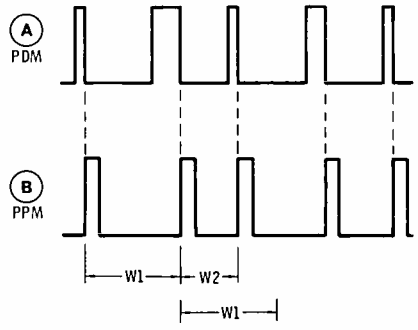


Fig. 8-10. Example of ppm using pdm as intermediary.

A pulse is generated from the trailing edge of each pdm pulse so that the relative positions of the pulses in Fig. 8-10B vary in accordance with the original signal amplitude. All of the pulses in a ppm signal are of the same amplitude and duration. Only the pulse positioning relative to each other (spacing between successive pulses) carries the signal information.

8-7. PULSE-CODE MODULATION (PCM)

The process of *pulse-code modulation (pcm)* is more complex than other pulse-modulation methods. It is, however, the most practical means for digitized audio and video signal handling and transmission.

Pulse-code modulation is achieved by sampling the analog signal, usually at a regular rate, and converting each instantaneous sampled level to a corresponding binary number. Since a certain amount of time is required to derive a binary number from an instantaneous sample, a sample-and-hold circuit is employed. This will be described later.

Quantizing

Fig. 8-11 shows the analog-to-pam-to-pcm function for an analog "ramp" signal. Each instantaneous sampled voltage level is held momentarily by the sample-and-hold circuit. The method is termed *quantizing*. The amplitude of the quantized signal in this example changes in equal steps and is a pam signal. Each step is termed a *quantum*, and the maximum quantizing error is $\pm\frac{1}{2}Q$. The midpoint of each step is the actual sampling time, and the steps simply indicate that the quantizing error can be plus or minus one-half quantum, which is the interval between samples. The midpoint of each step is then converted to a binary group as indicated. This is now a pcm signal, with a 4-bit binary number and 16 possible levels ($2^4 = 16$).

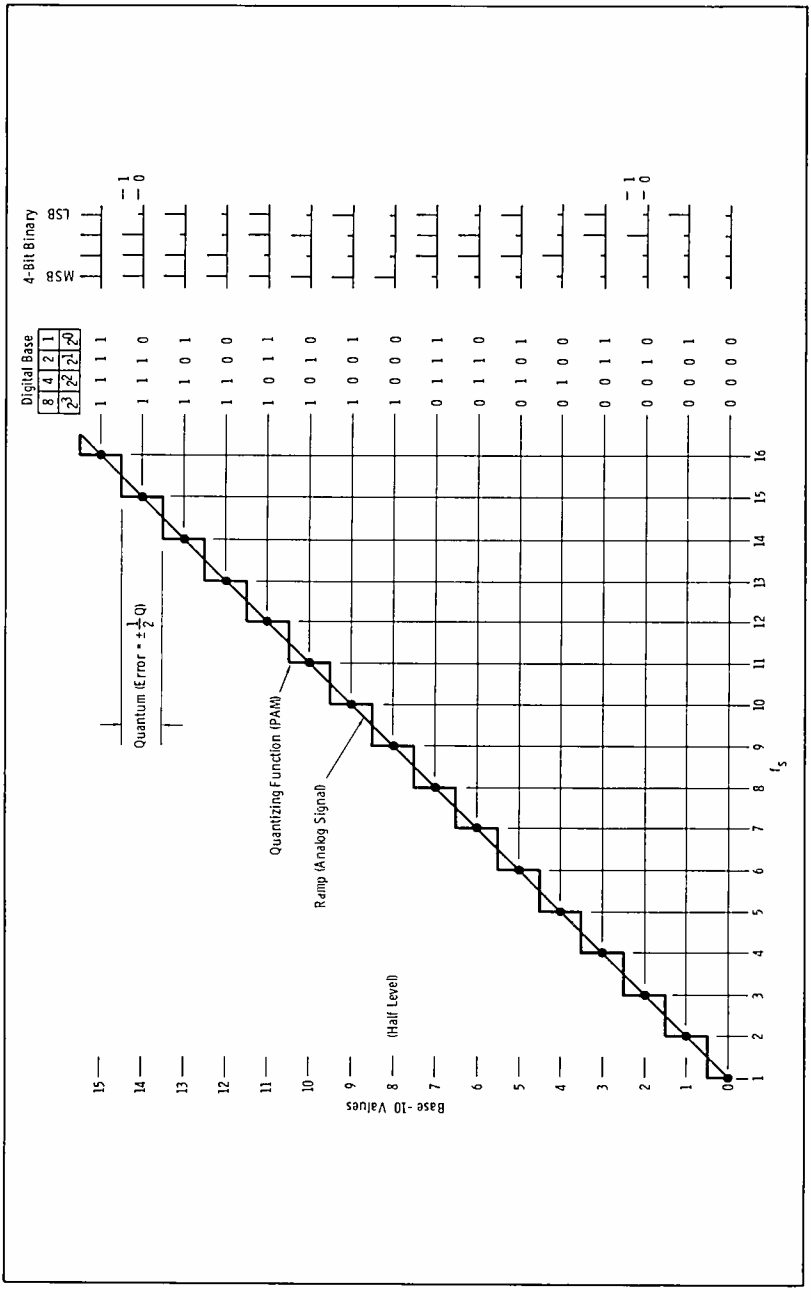


Fig. 8-11. Analog-to-pam-to-pcm conversion.

If this quantized signal (pam) is applied to a video monitor, steps (contours) of 16 shades from black to white will appear, rather than the continuous transition of the original ramp signal. Note, however, that if the quantized signal were filtered out before monitor application, the appearance would very closely approximate that of the original ramp signal.

Note also that the most significant bit (MSB) of the binary group first occurs at half-level. This understanding is useful in certain binary operations.

At the far right of Fig. 8-11, small pulses representing binary 0 are shown. In practice, typical values are binary 0 = 0.2 V and binary 1 = 4.0 V. This gives a voltage ratio of 4/0.2 = 20 = 26 dB, and 1's are readily distinguishable from 0's. For higher noise level transmission, the 1's may undergo further amplification to obtain a higher s/n power ratio.

The maximum quantizing error for the example in Fig. 8-11 is obtained as follows. Since there are 16 levels:

$$\begin{aligned} 1 \text{ quantum (Q)} &= 1/16 \\ \tfrac{1}{2}Q &= 1/32 \text{ or 1 part in 32} \\ &= 0.03125 = 3.125\% \text{ max quantizing error} \end{aligned}$$

Thus, if the ramp signal represented 0-100 IEEE units (0.714 V) on a video waveform-monitor graticule, then:

$$0.714 \times 0.03125 = 0.022$$

and:

$$\begin{aligned} 0.714 - 0.022 &= 0.692 \text{ min} \\ 0.714 + 0.022 &= 0.736 \text{ max} \end{aligned}$$

Another way of saying this is that the quantized signal (for 16 levels) is almost 97% accurate, or that 97% of the original information is contained in the quantized signal. The inverse of quantization accuracy is *quantization noise*. Quantization error and quantization noise are generally taken as one and the same.

For digital video systems (Chapter 10), 256 levels are used. To obtain 1/256 as the LSB, an 8-bit binary word is required ($2^8 = 256$).

The maximum quantizing error for a 256-level (8-bit) system is:

$$\begin{aligned} Q &= 1/256 \\ \tfrac{1}{2}Q &= 1/512 \\ &= 0.00195 = 0.195\%, \text{ or approx } 0.2\% \\ &\text{Quantizing accuracy} = 99.8\% \end{aligned}$$

The effect on a 0.714-volt signal is:

$$0.714 \times 0.002 = 0.0014$$
$$0.714 - 0.0014 = 0.7126 \text{ min}$$
$$0.714 + 0.0014 = 0.7154 \text{ max}$$

This error could not be read on the IEE scale.

Sampling theory, based on the Nyquist criterion (Section 8-3), says that the sampling (encoding) frequency must be at least twice the highest signal frequency concerned. In television, the highest signal frequency is normally taken to be 4.2 MHz. Thus, the minimum sampling frequency (f_s) must be at least twice the highest video frequency (f_v), or:

$$f_s = 2f_v = 2(4.2 \text{ MHz}) = 8.4 \text{ MHz}$$

In practice, to avoid phase distortion and filtering problems, $1.2(2f_v)$ is considered the minimum practical sampling frequency for digital video. Thus:

$$1.2(2f_v) = 2.4f_v$$

Since the highest f_v is 4.2 MHz, the minimum sampling frequency is $2.4(4.2 \text{ MHz}) = 10.08$ MHz.

Digital Video

Experimentation in digital video has shown that it is desirable that f_s be an odd multiple of one-half the line rate. A convenient choice is three times the color-subcarrier frequency (f_{sc}), or $3(3.58 \text{ MHz}) = 10.74$ MHz. This is a very common f_s at the time of this writing. However, some systems have used an f_s that is four times f_{sc}, or 14.3 MHz. This is an even multiple of the color-subcarrier frequency.

A high signaling speed is required in a digital video system. The 8-bit character for each of the 256 levels must be sent at three or four times the color-subcarrier frequency, depending on system choice. Thus, the signaling speed for the two sample frequencies is:

$$\text{At 10.7 MHz: } 8(10.7) = 85.6 \text{ megabits/second}$$
$$\text{At 14.3 MHz: } 8(14.3) = 114.4 \text{ megabits/second}$$

Sample-and-Hold Circuitry

Fig. 8-11 shows that each sampled level is "held" momentarily so that the pam signal can be converted to a system of binary numbers (pcm). Fig. 8-12 shows that the sample-and-hold circuit acts as a *data gate* controlled by the clock pulses. With Q1 and Q2 both cut off, all diodes are conducting from the current through R1 and R2. Thus, the data input charges the memory capacitor during this

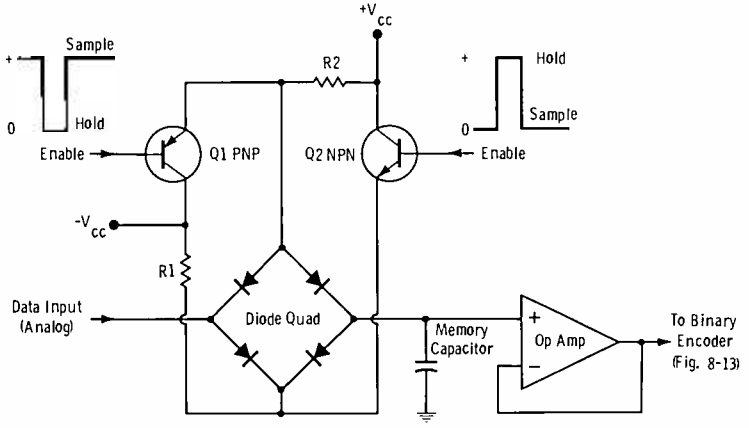


Fig. 8-12. Example of sample-and-hold gate.

time. (The gate is open, allowing passage of the input data.) This interval is the sampling time.

When the clock pulses (opposite polarity in this example) are applied, the diodes become reverse-biased, closing the gate so that the charge on the capacitor is "held" at the level sampled. The operational amplifier acts as a buffer between the memory capacitor and the encoder input.

PAM to PCM

Several different means are available for converting pam signals to pcm signals (review Chapter 7). One type (successive approximation) is shown in Fig. 8-13. This illustrates memory for a 4-bit (16-level) binary.

At the start of each sampling interval, each flip-flop (FF) is set by a clock pulse. Note that the clocked inputs are to both the AND gates and the flip-flops. Clock pulses occur simultaneously, speeding up the conversion; *i.e.*, 2 and 3, 4 and 5, and 6 and 7 are simultaneous, and 8 and 1 are the "start" pulse.

As noted previously, the MSB of any binary word first occurs at half level. The MSB responsible for half-scale voltage sets the action of FF1 in Fig. 8-13. Its weight is converted back to an analog signal by the dac. If this input to AND gate 1 (via the difference amplifier) equals or exceeds ½, the bit is allowed to remain. If it is less, the MSB is turned off by the clock pulse at 2.

The process is repeated for the next MSB (¼ voltage level) until the LSB (1/16 level) has been subtracted or added. Thus, a 4-bit binary character has been created from the instantaneous voltage level sampled.

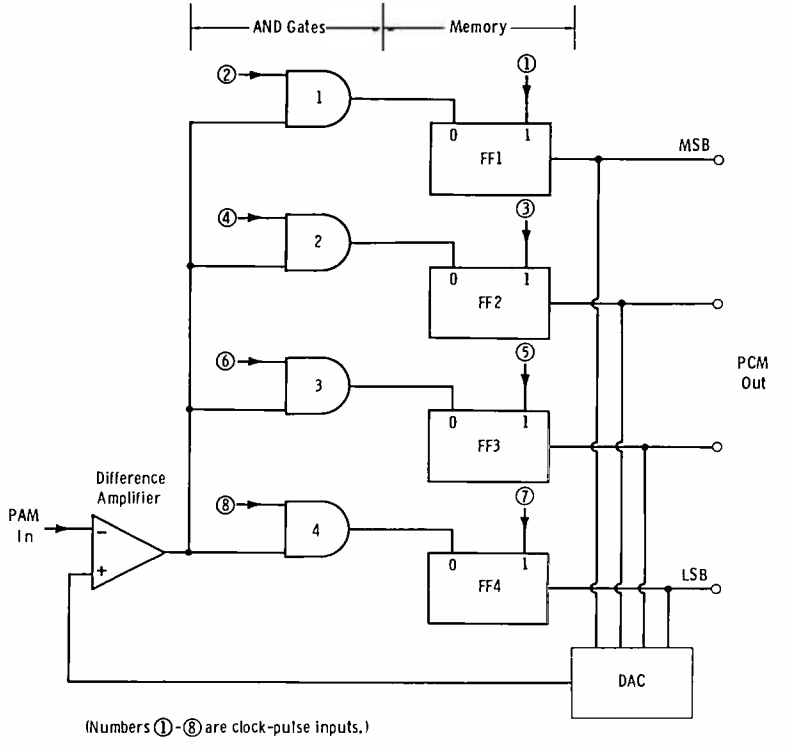


Fig. 8-13. One type of pcm-to-pcm encoder (successive approximation).

Obviously, an 8-bit binary word requires eight memories. This means that an 8-bit group is created for each of the 256 levels at a rate equal to three or four times the color-subcarrier frequency (for digital video systems). The associated successive-approximation circuitry would require eight "looks" for this 8-bit system. For this reason, higher-speed methods of conversion, such as those outlined in Sections 7-5 and 7-7 of the previous chapter, are generally used for digital video systems. Also review dac methods in Section 7-9 for the final step in a digital system.

8-8. MULTIPLEXING

Multiplexing allows simultaneous transmission over the same path of two or more signals in either or both directions. A common application in broadcasting is in remote operation of transmitters. Data are sent to the transmitter from the studio for remote control of operating parameters, and data are sent back from the transmitter

to the studio to indicate these operating parameters. The data returned from the transmitter may also be used for automatic logging.

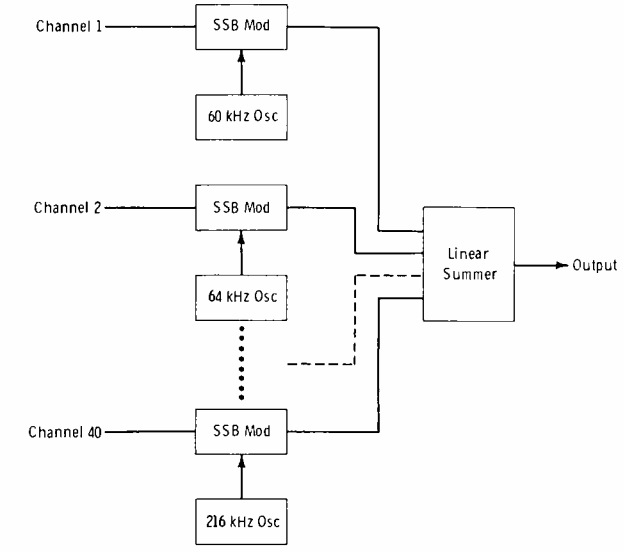
The two basic types of multiplexing are:

1. Frequency-division multiplexing (fdm)
2. Time-division multiplexing (tdm)

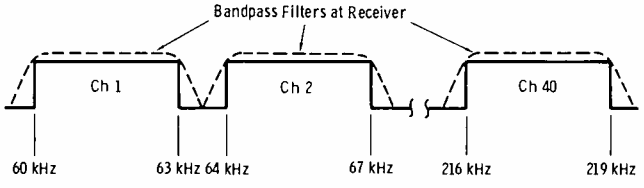
Until recently, fdm carried the major portion of long-haul communications traffic, and it will be considered first.

Frequency-Division Multiplexing

Fig. 8-14A shows a representative voice-band fdm transmitter, and Fig. 8-14B shows the frequency allocation. Each channel is band-limited to 3 kHz, and there are 40 channels. Each signal channel is used to modulate a carrier 4 kHz higher in frequency than the preceding carrier. Only the upper sideband is used. The modulator outputs are fed to a linear summer where they are added. The summer



(A) Block diagram.



(B) Frequency allocation.

Fig. 8-14. Forty-channel voice-band fdm system.

output is a series of channels 3 kHz wide and separated by 1-kHz guard bands. In this example, a frequency spectrum of 60 kHz to 219 kHz is utilized. The 40 channels are carried simultaneously over a single transmission path.

At the receiving terminal, each channel is extracted by an appropriate bandpass filter and then mixed in a product detector with the appropriate carrier frequency. This is the same carrier frequency that was used to generate the ssb signal at the transmitter terminal.

With fdm, each information channel occupies a small percentage of the transmission spectrum, and extreme linearity is required. With nonlinearity, harmonics of the 60-kHz channel will appear on higher channels, resulting in crosstalk. Cross-modulation products can also occur; the 60-kHz and 64-kHz channels can mix, creating noise in the channel at 124 kHz, etc.

Time-Division Multiplexing

With tdm, the *time* of transmission rather than a *frequency* band is shared. Because the tdm signal occupies the entire spectrum during the time of transmission, cross talk and cross-modulation distortion are not bothersome. The time-division process means that each information channel is periodically sampled, and these samples are sequentially interlaced into a composite baseband signal. For analog transmission, the samples themselves may form the baseband signal. If digital transmission is used, the samples may first be converted to code groups (blocks) which are then transmitted.

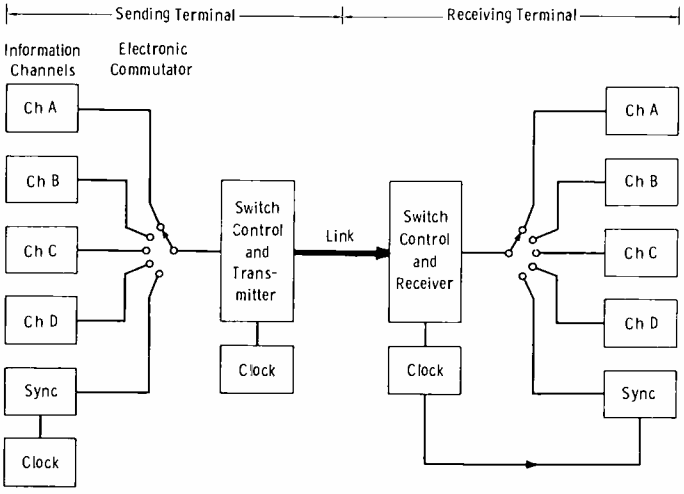


Fig. 8-15. Simplified block diagram of tdm modem.

Fig. 8-15 gives the basic idea of a tdm *modem* (modulation-demodulation system). At the sending terminal, an electronic switch acts as a commutator that momentarily selects a given channel of information. Thus, a specific channel is transmitted during a specific time slot. When all the channels have been sampled once, a *frame* has occurred. Frames occur sequentially and correspond to the order of channel sampling. Therefore, some form of synchronization is required in any tdm system. Synchronization permits the receiver decommutator (demultiplexer) to select the specific channel being sampled at a given time. The clock at the receiving terminal in Fig. 8-15 is derived from the sync channel at the transmitting terminal.

Fig. 8-16 shows a simplified diagram of tdm. Nine information channels and one sync channel are shown. The data signals may be analog or digital. The gates may be simple AND circuits. Although

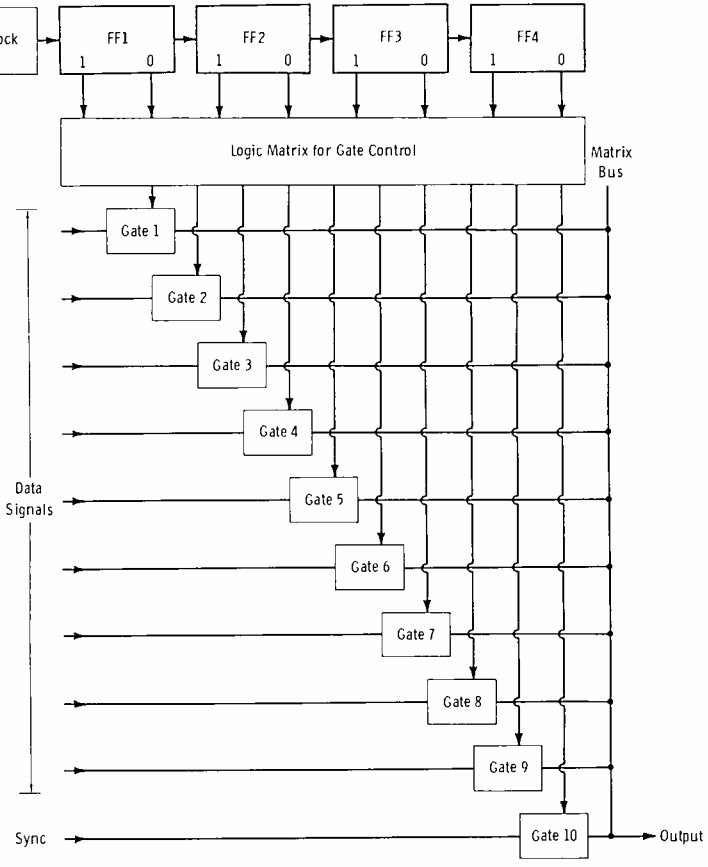


Fig. 8-16. A simplified diagram of tdm.

a series of flip-flops is shown in Fig. 8-16, tapped delay lines may also be used to obtain the delayed gating pulses.

To get a clear picture of tdm, look at it this way: Assume a voice channel with bandwidth to 3 kHz and a sampling rate of 8 kHz, which is well above the Nyquist minimum of 6 kHz. Further assume that each sampling pulse is allowed a duration of 1 μs. Since the sampling rate is 8 kHz, pulses for this one channel are $1/(8 \times 10^3) = 125$ μs apart. This means that there is time available for 124 additional 1-microsecond channels (bandwidth to 3 kHz) that could be transmitted over the same path (Fig. 8-17). Note that in this case the pulse format required would be NRZ. Unless one of the adjacent information channels happens to be zero at the sampling instant, the pulses are touching side by side, requiring instantaneous switching to avoid information overlap. Therefore, an RZ sampling format is generally used, as illustrated in Fig. 8-18. In this example, each channel pulse is 1 microsecond long and there is a 1-microsecond interval between pulses. This arrangement cuts the number of available channels to one-half that of an NRZ system.

The sync pulse (for other than pcm) is generally of higher amplitude and/or greater width than the informational-channel pulses. This allows the receiver to distinguish the synchronizing information from the data information. In Fig. 8-18, the sync pulse is 3 μs wide. The trailing edge of sync resets the commutator to Channel 1 and is therefore the "start" information. For pcm, the sync signal is a certain unique code and is not dependent on amplitude or width.

Note that the example of Fig. 8-18 allows 61 channels. The number of channels could be doubled and the RZ format retained by using $\frac{1}{2}$-μs pulses separated with $\frac{1}{2}$-μs intervals. This would allow 122 channels (plus sync) over the same link.

The clock rate must be at least equal to the sampling rate times the number of channels being sampled, including the sync channel. If the sampling rate is 8 kHz and the number of channels is 125, then the lowest clock rate would be:

$$(8 \times 10^3)125 = 1 \text{ MHz}$$

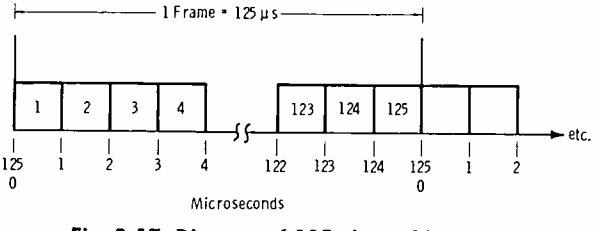


Fig. 8-17. Diagram of 125-channel NRZ tdm.

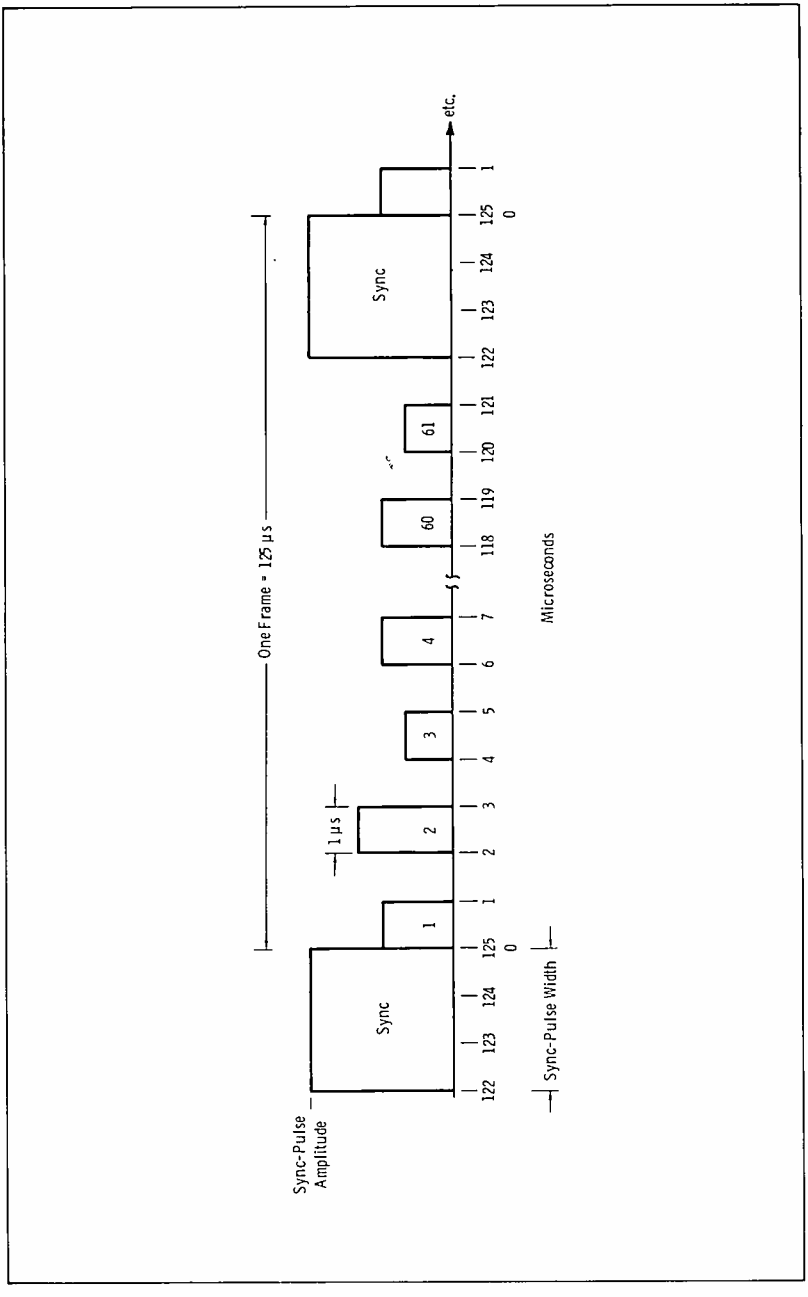
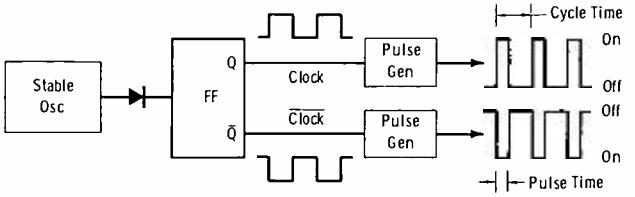
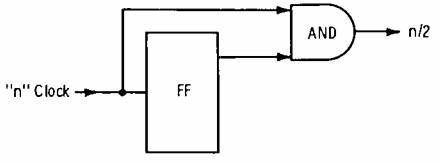


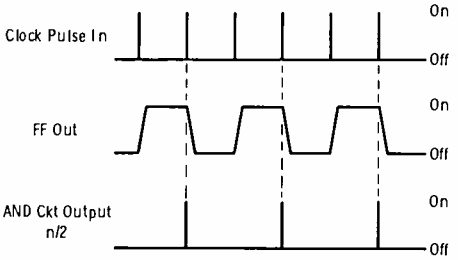
Fig. 8-18. Practical 61-channel pam, RZ-pulse transmission frame.



(A) Clock-pulse generator.



(B) Clock-pulse divider.



(C) Waveforms.

Fig. 8-19. Basic clock-pulse generator and division process.

8-9. CLOCK GENERATORS

A circuit arrangement termed a *master clock* provides the main timing signals for a digital unit. A stable oscillator (Fig. 8-19A), which may provide either a sine-wave or a square-wave output, is used as the basic timing reference. This drives a flip-flop that provides high (clock) and low (clock) outputs. Both outputs may be used in a system, depending on polarity requirements.

The clock signal is normally composed of rectangular pulses rather than square waves. Therefore, pulse generators are required to give a narrower pulse relative to the cycle time as shown at the output of Fig. 8-19A. These may be simple "boxcars"[1] or other types of pulse generators.

---

1. Harold E. Ennes, *Workshop in Solid State* (Indianapolis: Howard W. Sams & Co., Inc., 1970), pp 205-209 and 240-246.

A clock frequency that is an exact submultiple of the master frequency is often required. The bistable flip-flop is a natural divide-by-two circuit and is normally used to obtain frequency division (Fig. 8-19B). The master clock is applied to one input of an AND circuit. The output of the toggling flip-flop feeds the other input. The ANDed output occurs only at the coincidence of high levels, as shown by Fig. 8-19C. Thus, the output is one-half the master clock frequency. If more division is required, additional flip-flops are used.

In some systems, multiple-phase clock signals are employed. Fig. 8-20 shows a typical timing diagram where additional phases are obtained from the trailing edge of each clock pulse. The cycle time is equal to four times the phase time.

8-10. RANDOM AND IMPULSE NOISE

The random noise level of any given link is a predictable number as found from periodic s/n-ratio measurements over the link in question (Fig. 8-21A). Thus, the required signal power may be predicted and used to provide an acceptable s/n ratio.

Impulse noise (Fig. 8-21B) is not predictable in most cases. This form of noise is the largest source of data-communication errors.

Interlacing is a term used to describe a technique that is similar to time multiplexing and is used to counteract the effects of impulse noise. The following steps refer to Fig. 8-22:

A. Five words (of five bits each) in this example are read into a memory device such as a flip-flop register. The read-in is *horizontal,* and the words represent 10101.
B. The five words are scanned *vertically* and transmitted over the link in this sequence. The information is read out, but not as

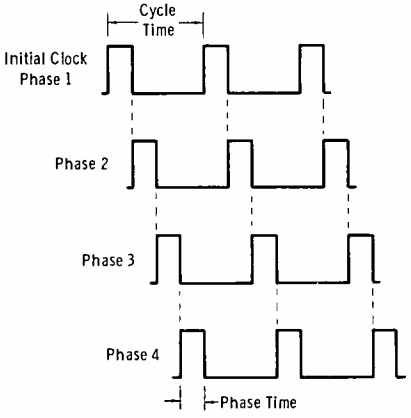


Fig. 8-20. Timing sequence for four-phase master clock.

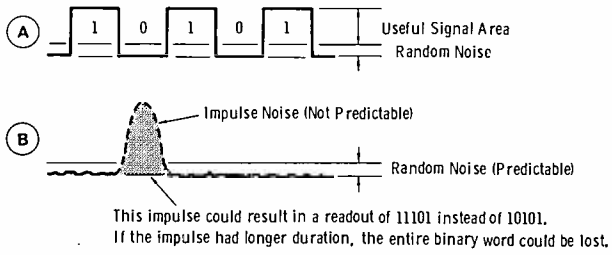


Fig. 8-21. Random and impulse noise levels.

it was stored. The first bits from each of the five words are sent together, then the second bits from each of the five words, and so on until all bits are sent. Now assume that an impulse occurs that obliterates word three.

C. The received data now appear in this fashion. All the error bits are in one word.

D. The vertical scanning of (C) constructs five words with one error bit each. Thus, instead of one entire word being obliterated, only one error bit occurs in each word. If the code has sufficient redundancy (such as parity bits), the original words 10101 can be easily reconstructed.

This is a very effective technique when multiple parity bits are used so that error detection and error correction are both possible at the receiving end. The method is more appropriate for synchronous transmission than for asynchronous (nonsynchronous) transmission.

EXERCISES

Q8-1. Define the term *data link.*

Q8-2. Define (A) on-line operation (B) off-line operation.

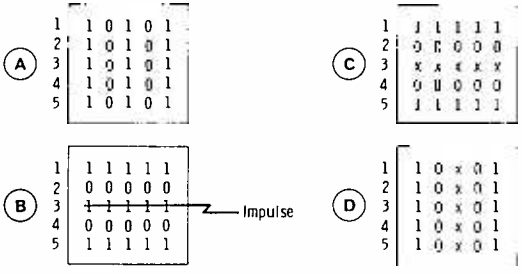


Fig. 8-22. The technique of interlacing.

Q8-3. Give three basic classifications of data links according to data character.

Q8-4. Give three basic classifications of data links according to useful direction of transmission.

Q8-5. What does the term *data* mean?

Q8-6. Give two basic reasons for modulation.

Q8-7. Calculate the minimum sampling rate for an information channel that extends from 29 kHz to 32 kHz.

Q8-8. Give the two most common types of multiplexing.

Q8-9. A telemetry link uses pam to multiplex ten channels. (A) If the highest expected frequency in any channel is 10 Hz, what is the maximum theoretically possible frame time? (B) Assuming the interval between pulses is equal to the pulse duration, what is the maximum possible pulse duration? (Disregard synchronizing pulses.)

Q8-10. Why must a sample-and-hold circuit be used with pcm?

CHAPTER 9

Digital Control Systems

This chapter will consider the general control aspects of digital systems. Such applications are:

1. Remote transmitter controls
2. Audio and video source logic switchers
3. Audio and video tape sync and editing systems
4. Digital character generators
5. Digital synchronizing generators
6. Digitally controlled color cameras

This type of digital system is used to *control* various analog devices such as transmitters, recorders, etc. The actual conversion of analog signals to digital form for further processing and control is an entirely different field, which is covered in Chapter 10.

9-1. REMOTE TRANSMITTER CONTROL

Earlier nondigital remote transmitter control systems generally employed different frequency tones for specified control purposes. These tones were carried either by landline or by a subcarrier on a radio link. Such tones are subject to modification as a result of stray pickup or other types of interference.

A digital remote-control system may also employ tones. The big difference is that the tones are formed into unique binary codes so that bandpass filters (where employed) do not depend upon a fixed frequency for proper operation, but more upon a *frequency ratio* property. For a given system, a certain frequency represents a 0, and another specified frequency represents a 1. Furthermore, a specified code consisting of (for example) seven bits and one parity bit is used to address a given section of the transmitter and to perform a

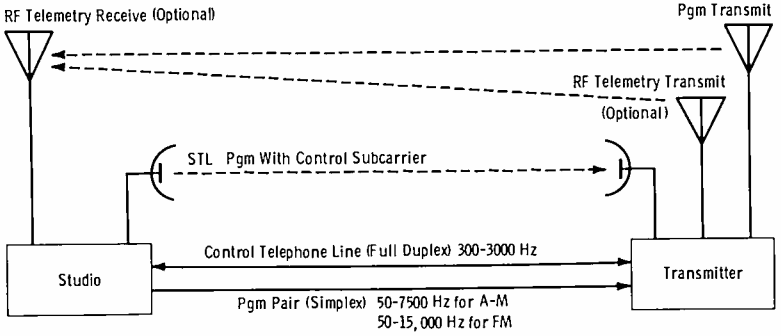


Fig. 9-1. Transmitter remote control (both landline and stl facilities shown).

specified function in that section, so that extreme reliability can be achieved.

For example, assume a 6-bit byte. The first five bits could select any of $2^5 = 32$ functions. These functions are normally final plate voltage, final plate current, operating frequency, audio gain, etc. This leaves one bit of the 6-bit byte for a command (control) function. One bit provides $2^1 = 2$ control parameters, which are normally "raise" and "lower." Parity bits are often used to assure accuracy of the command.

Fig. 9-1 illustrates the basic idea of remote transmitter control from the sending (studio) location. When Telco (telephone line) facilities are used, the control line is in addition to the normal program-transmission pair. Unlike the program line, the control pair need have only voice-band frequency response, typically 300 to 3000 Hz. For most digital systems, dc continuity is not required. This line must be full duplex, since command functions are sent to the transmitter and remote-metering signals are returned to the studio.

In general, command signals lie in the 2000–2500 Hz range. For frequency modulation, 2300 Hz could represent binary 1, and 2500 Hz could represent binary 0. The return metering information is normally provided by telemetry in the 1200–1500 Hz range, or by subaudible telemetry in the 18–37 Hz range. When landlines are used with subaudible telemetry, a carrier of around 1300 Hz is amplitude- or frequency-modulated by the telemetry signal to provide proper transmission above 300 Hz.

The metering information provided by telemetry is generally obtained from sampling resistors included by the transmitter manufacturer in the circuits to be remotely controlled. A sampling dc voltage (normally in the 1-to-10 volt range) is converted to the telemetry frequency by means of a dc-to-ac converter. For example,

if 1 volt dc is assigned to represent the normal desired operation, 2 volts dc would indicate a level 100% higher than normal. A linear voltage-controlled oscillator is used to convert the varying dc voltage to a corresponding varying frequency. This varying frequency is normally in the form of square waves, as from a multivibrator circuit.

When the communications system employs stl's rather than landlines, the command and telemetry signals are carried on subcarriers. For example, an a-m or fm stl operating in the 950-MHz band may have subcarriers of 26 kHz, 67 kHz (normally used for SCA in fm broadcasting), or a frequency between 110 and 185 kHz. When both the stl and binary code are fm, this becomes an fm/fm system. The return path for the telemetry information can be either of the SCA channels of an fm transmitter, or it can be a separate rf link, as shown in Fig. 9-1. Subaudible telemetry return can be over the main channel of an a-m transmitter (fm/am system) or by an optional rf link.

Frequency-Shift Keying (FSK)

For pulse applications, a specific type of frequency modulation called *frequency-shift keying* (*fsk*) is used. With this method, one frequency represents binary zero, and another frequency represents binary one.

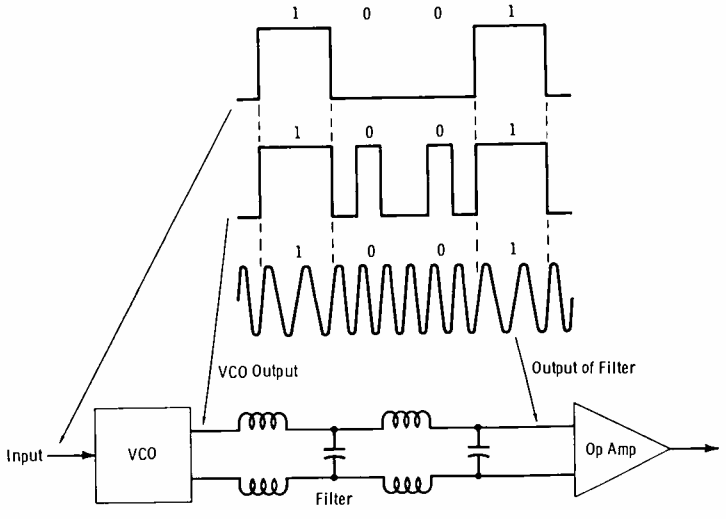
A simplified diagram of an fsk generator is presented in Fig. 9-2A. This may be an astable multivibrator with a reference dc base voltage that results in a given frequency of operation. The point at which the reference base voltage is intercepted is governed by an applied signal dc, changing the frequency of operation.[1] For binary applications, only two frequency values (0 and 1) result. For telemetry, the range of output frequencies must be linear over a given range of dc voltage values employed to operate LED display devices.

As shown by Fig. 9-2A, the output of the generator is filtered to provide corresponding "tones" that can be transmitted via wire or radio relay. As shown by Fig. 9-2B, the fsk demodulator normally employs a phase-locked loop so that the output signal is under tight control of the input signal.

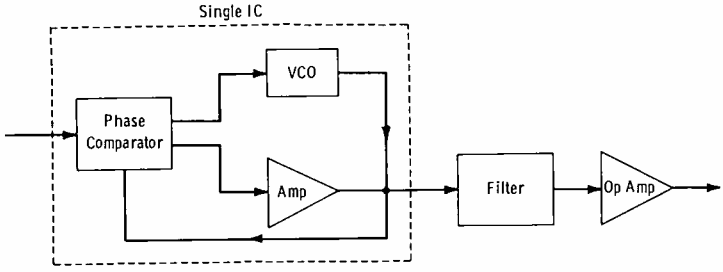
Phase-Shift Keying (PSK)

For binary applications in which a single frequency may be desired, *phase-shift keying* (*psk*) may be encountered. Fig. 9-3 shows binary information for which a single frequency is used, but a 180° shift in phase occurs at the leading and trailing edges of a pulse. The

1. Harold E. Ennes, *Television Broadcasting: Systems Maintenance* (Indianapolis: Howard W. Sams & Co., Inc., 1972), pp 195–196.



(*A*) *Generator.*



(*B*) *Demodulator.*

Fig. 9-2. Simplified diagrams of basic fsk modem.

frequency may be subaudible, audible, supersonic, or rf. The signal is demodulated by a simple phase discriminator.

The Moseley DRS-1 Digital Remote System

The Moseley Model DRS-1 supplies totally digital command, telemetry, and status/alarm functions and automatic parameter logging as an optional feature. Interconnection between the control terminal and remote terminal can be accomplished by either wire or wireless means.

For landline operation, a two-wire, voice-grade (300–3000 Hz), 600-ohm balanced telephone circuit is required. Maximum permissible attenuation is 30 dB when the send level is 0 dBm. All functions are accomplished via fsk signals in the following frequency ranges:

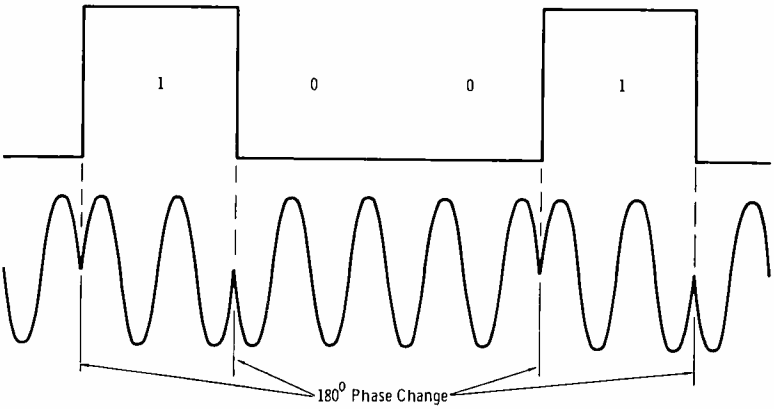


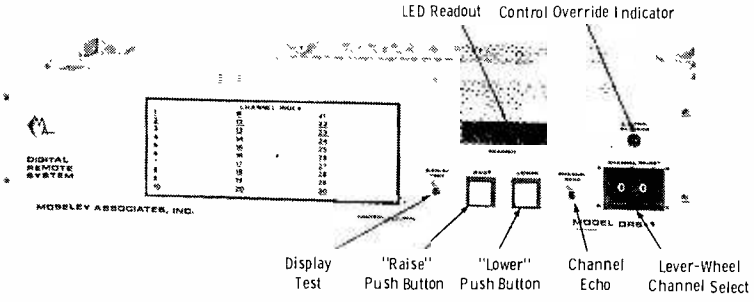
Fig. 9-3. Phase-shift keying (psk).

Command (control): 2300–2500 Hz
Telemetry (metering): 1300–1500 Hz
Digital logging system (optional): 750–900 Hz

For wireless service, control information is transmitted to the remote terminal via an fm subcarrier on the stl. For monaural stl's, a 26-kHz subcarrier frequency is normally utilized. For a composite (stereo and SCA) stl system, a subcarrier frequency of 110 kHz is available. Audible telemetry (1300–1500 Hz) is standard. Subaudible telemetry (18.75–37.5 Hz) is available on special order for return via an SCA channel of an fm transmitter.

A view of the front panel of the control terminal is shown in Fig. 9-4. Fig. 9-5 is a simplified block diagram of the complete control system.

The basic digital remote system is divided into three units—control terminal, remote terminal, and selector unit(s). The control terminal is located at the remote control point, normally the studio



Courtesy Moseley Associates, Inc.

Fig. 9-4. Front-panel view of control terminal.

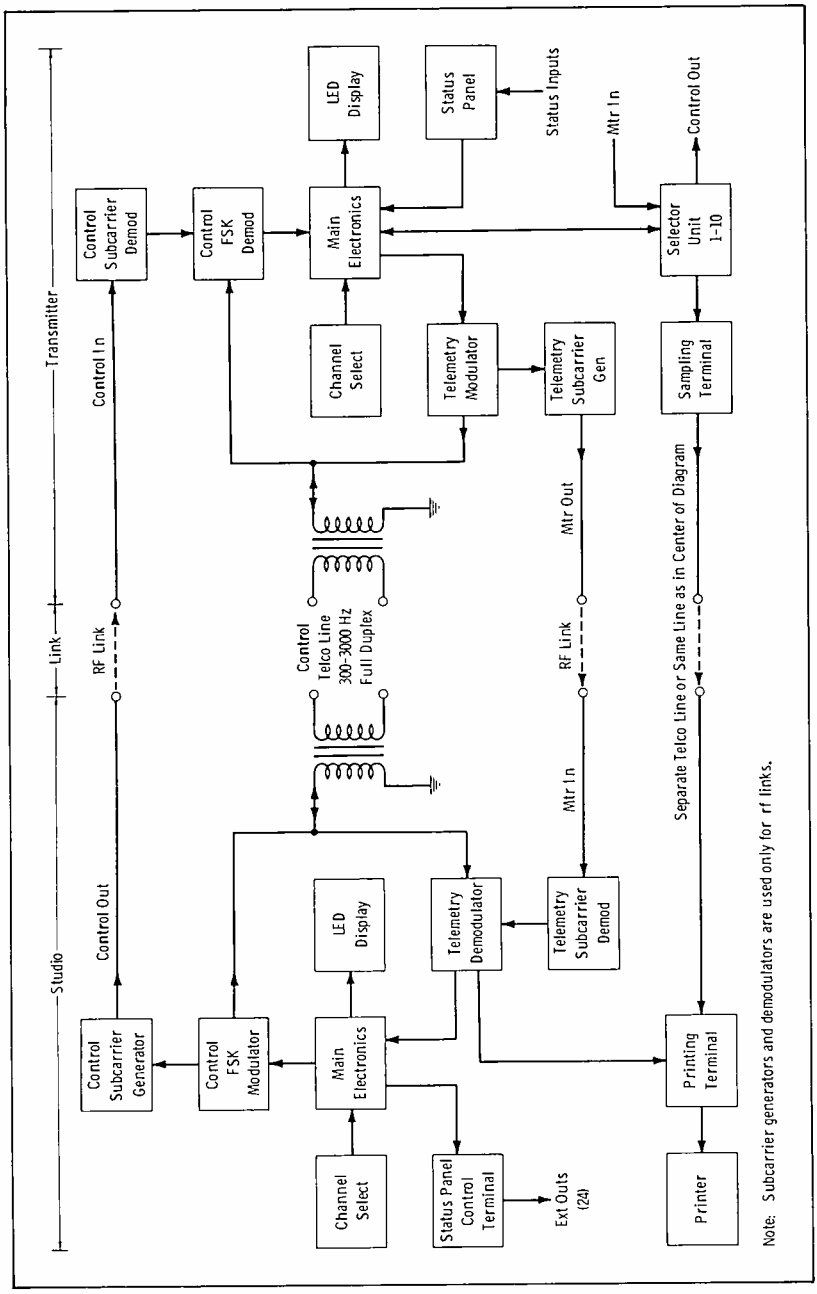


Fig. 9-5. Simplified block diagram of digital remote system.

location in broadcast-transmitter remote control. The remote terminal and selector units are placed at the transmitter site, or the location of the equipment being controlled. Positioning can be seen in the block diagram of Fig. 9-5. Each selector unit provides ten telemetry/command channels, which allows for field expansion or tailoring to fulfill specific channel requirements. A maximum of 30 channels (three selector units) may be used.

Each telemetry/command channel provides a single telemetry function and two command functions. These command or control functions are individual Form A, isolated dry contact closures and are typically identified as "raise" and "lower." The raise and lower command outputs can switch external loads of up to 50 watts, noninductive at potentials of 120 volts ac or dc. Telemetry inputs accept a dc sample voltage representing the desired analog parameter. This dc voltage is typically in the range of 1 to 10 volts.

Every telemetry channel is displayed digitally as a 3- or 3½-digit numeric presentation. With an input of 2 volts dc, a maximum numeric display of 1999 is possible. A display of 999 can be obtained with an input of 1 volt dc. Individual calibration potentiometers, provided on each telemetry channel, permit the calibration of the numeric display to be identical with the value of the selected parameter. Digital numeric light-emitting diode (LED) displays are located on both the control and remote terminals. One-man system calibration is afforded by the digital numeric display on the remote terminal. For calibration, it is only necessary to adjust the calibration potentiometer until the digital numeric display shows a value equal to the selected parameter.

The front panel of the control terminal (Fig. 9-4) contains channel-select and command controls as well as a telemetry display. Telemetry/command channels are selected with centrally located lever-wheel assemblies. These assemblies provide direct-reading channel numbers. After channel selection, the parameter being telemetered is automatically displayed. Raise and lower buttons permit activation of command functions. When the command function is accomplished, the raise or lower button is illuminated as a true-tally, signifying activation of the raise or lower relay. The channel echo capability allows tally-back and display of the selected channel.

All controls and the telemetry display are duplicated on the remote terminal. Local control at the transmitter site is taken by the REMOTE/LOCAL switch. When this switch is depressed, command information, including channel selection, can be accomplished only from the remote terminal. In the local mode, an indication is sent to the control terminal, and the CONTROL OVERRIDE LED is illuminated to inform the operator that the command capability is not available

at his location. An additional set of contacts on the REMOTE/LOCAL switch is terminated on the rear apron of the remote terminal and can be used for an external indication of the switch position. These contacts are commonly referred to as "go home" indicators, and they can be used to help prevent maintenance personnel from leaving the remote terminal in the local mode.

Digital techniques are used for all functions. In the remote terminal, dc sample voltages are converted to serial digital information by an analog-to-digital converter (adc). Both telemetry and command information are organized into 8-bit serial digital words. Modems in the remote and control terminals permit transmission of the digital words. These modems convert the digital words to fsk audio tones that can be transmitted over telephone or radio circuits. Error checking is included to ensure accuracy of telemetry and command information.

Fail-safe circuitry complies with Federal Communications Commission requirements for broadcast transmitters. For the control fail-safe requirements of standard, fm, and tv service, the presence of correct command information is continuously sensed in the remote terminal. Should command information not be received for a period of 20 seconds, the control fail-safe relay in the remote terminal is deactivated. For telemetry fail-safe with television transmitters, the DRS-1A functions with the Model FSU-1 fail-safe unit. With an interconnection interruption or other telemetry failure, the digital numeric display flashes on and off at a 1-Hz rate. Further, an output is provided on the rear apron of the control terminal for an external alarm indication.

The status subsystem provides for 24 status/alarm (go/no-go) indications. Typically, applications include the sensing of illegal entry, fire and smoke alarms, over/under temperature, transmitter status, tower lights, or any similar go/no-go condition or function. The subsystem consists of the control-terminal status panel and the remote-terminal status panel. Both panels operate from the power supplies and modems in the DRS-1A control and remote terminals. Each status channel is encoded from external, normally open, dry contacts. Closing of these contacts illuminates a light-emitting diode (LED) on both status panels. The subsystem is of the latching type. When a channel is activated, the LED remains illuminated until it is manually reset. Resetting may be accomplished at either the remote or the control terminal.

An additional feature is provided by the control-terminal status panel. A transistor sink to ground is provided for each channel that is activated when a channel is encoded. These sinks can switch an external load of 200 mA, 24 volts dc, permitting the use of external alarms, relays, warning lamps, or similar devices.

9-2. AUDIO AND VIDEO LOGIC SWITCHERS

Digital switching systems are used in both audio and video systems, and they are similar in design for either application. Because of the extended frequency range of video signals (to 10 MHz), a compensating capacitance (described later) must normally be switched in and out across large switching buses to compensate for on or off conditions of a particular digital crosspoint. Otherwise, the designs are identical, except that a self-contained master clock-pulse generator is employed for audio switchers in lieu of a master clock driven from vertical drive or composite sync in television switchers.

Fig. 9-6 is a simplified block diagram of a typical digital switching system. Vertical drive (or composite sync) is employed to synchronize the switching pulses with the vertical-blanking interval. The switching pulses are generated by a one-shot multivibrator on the +5-volt regulator board; one regulator board supplies each bay of 12 logic cards, and each logic card handles six crosspoint controls.

A simplified schematic diagram of a typical video crosspoint (Fig. 9-7) illustrates the function of switching. Note that quad 1 acts as a gate which is either open or closed to the video input. We know that point A of quad 1 must be positive and point B must be negative in order for the video signal to pass to the output amplifier. Therefore, Q4 must be cut off (to supply a negative voltage to point B), and Q3 must be cut off (to supply a positive voltage to point A). Note that if Q4 and Q3 were saturated, the polarities would be reversed, and quad 1 would be off.

Note that quad 2 provides just the opposite off-on function from that provided by quad 1. Thus, when quad 1 is a closed circuit (on), quad 2 is an open circuit (off). But when quad 1 is opened (off), quad 2 is closed to the 100-pF capacitor so that the same capacitance is presented to the video signal bus as when the signal is connected to the input capacitance of the video amplifier.

Now let us analyze what input condition is necessary to close quad 1 to pass video. Since Q4 must be cut off, the base of Q4 must not be negative relative to its emitter (pnp transistor). Thus, we can see that Q1 must also be cut off. Since Q1 is an npn transistor, the junction of D1 and D2 must be essentially at ground so that the current through R_{B1} will not forward-bias the transistor. From this we note that a low level at the crosspoint logic input must exist to "fire" the crosspoint, and that a high level (positive to cut off D1) turns the crosspoint off.

Fig. 9-8 shows the function of the +5-volt regulator and one-shot board. The +10 volts from the main regulated power supply (which supplies +10 and −10 volts to the video amplifiers) is converted to +5 volts, regulated. Vertical drive (or a vertical-rate pulse generated

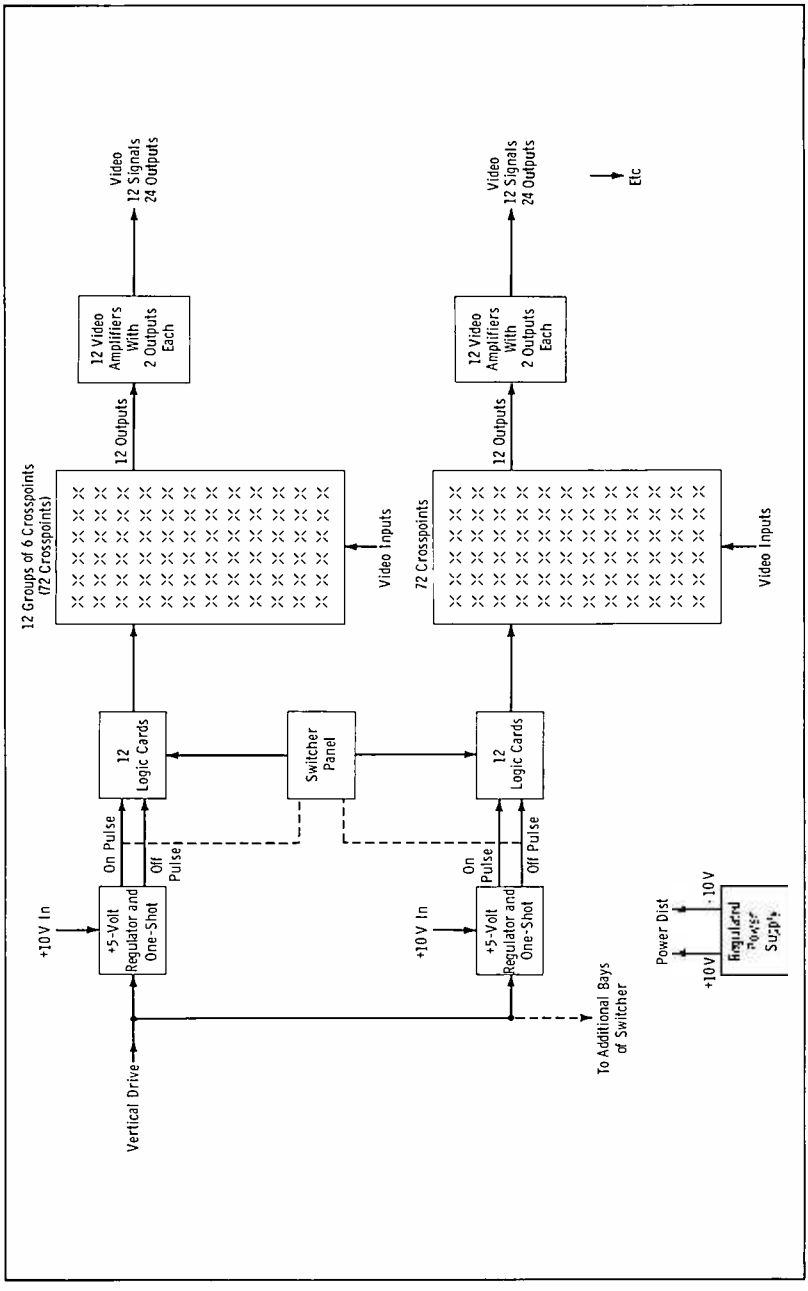


Fig. 9-6. Simplified block diagram of digital switching system.

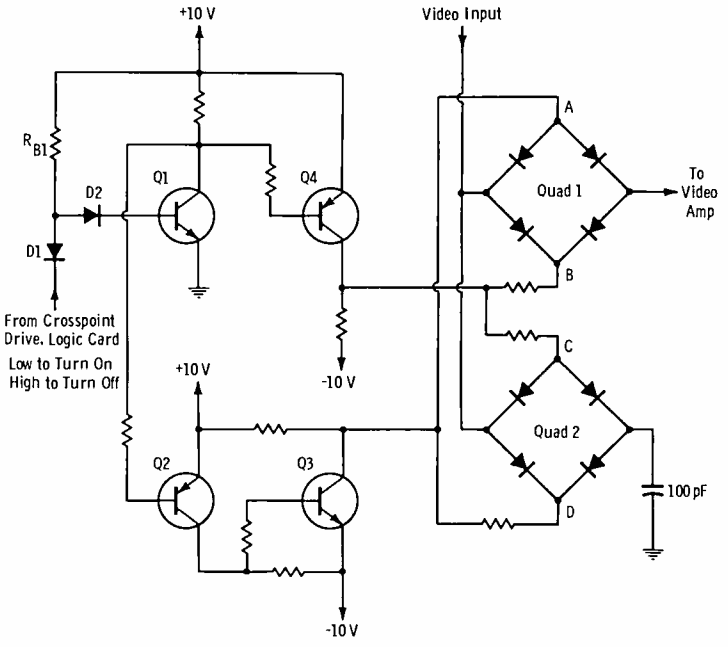


Fig. 9-7. Simplified diagram of video crosspoint in digital switcher.

from composite sync) is used to control two multivibrators. Multivibrator 1 supplies the on pulse, and multivibrator 2 supplies the off pulse which is much narrower than, but coincident with, the on pulse.

Fig. 9-9 is a simplified schematic diagram of a digital-switcher logic card for the purpose of functional analysis. Note that the control pulses from the one-shot multivibrators are applied continuously. The operating mode cannot be changed, however, until one of the push buttons on the switcher control panel is pressed to select a signal.

Assume that the switcher has just been turned on and no source has been punched up (switcher in black). Although the on pulse is being applied (60 times per second) to pin 2 of IC1, since this is a NAND gate, no change can occur unless pin 1 is also positive. Since pin 1 is grounded through R1, it is essentially at ground potential. The same conditions apply to IC2.

Now assume that the switch for source 2 is depressed momentarily. This applies +5 volts to pin 1 of IC2, and when the positive pulse arrives at pin 2, a negative pulse is produced at pin 3. Prior to the switching sequence, pin 8 is at the high level, and this level is coupled to pin 5. Therefore, until the negative-going pulse from pin 3

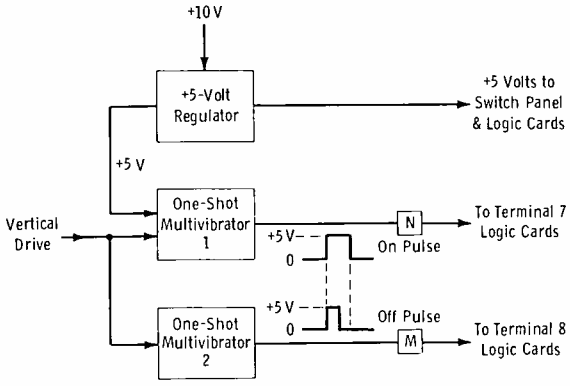


Fig. 9-8. Simplified block diagram of 5-volt regulator and one-shot.

is applied to pin 4, both pins 4 and 5 are at the high level, and pin 6 is at the low level. When the pulse appears at pin 4, pin 6 goes to the high level, causing both pins 9 and 10 to be at the high level (as soon as the narrow off pulse at pin 9 terminates). Pin 8 then goes to the low level; pin 5 also goes to the low level, latching the gates in this state. Since the drive for the crosspoint is obtained from pin 8, the input of the crosspoint is now at the low level, and therefore the crosspoint is turned on (Fig. 9-7).

Now we have signal source 2 on the air and desire to put source 1 on the air. With button 1 depressed, the action described for the IC2 path is repeated for the IC1 path. Since no positive potential is applied to pin 1 of IC2, the narrow off pulse at pin 9 of IC2 will reverse the state of that gate and turn crosspoint 2 off when crosspoint 1 is turned on. Note that the gate on IC2 with pins 11, 12, and 13 can be activated only when a switcher-panel button is pressed to select a source, applying a coincidence pulse to pin 13 of IC2. Thus, a crosspoint cannnot be turned off until another source is turned on.

Note that when source 1 is turned on, diode D1 becomes forward-biased. This turns on Q1, which turns on Q2 to complete the circuit of indicator lamp L1 for indication that the associated crosspoint is on. These transistors are termed *lamp drivers*. When the gate is turned off, the diode becomes reverse-biased, Q2 becomes an open circuit, and the lamp is extinguished.

Preprogrammed studio switchers (Fig. 1-6, Chapter 1) change audio and video sources, provide previews of upcoming video sources, preroll film projectors and video tape machines, start audio tape units, stop projectors and video tape systems, change slides, switch multiplexers, etc. In addition, some expanded automation systems display the remaining duration of the on-air event along with its video and audio sources; display scheduled duration and

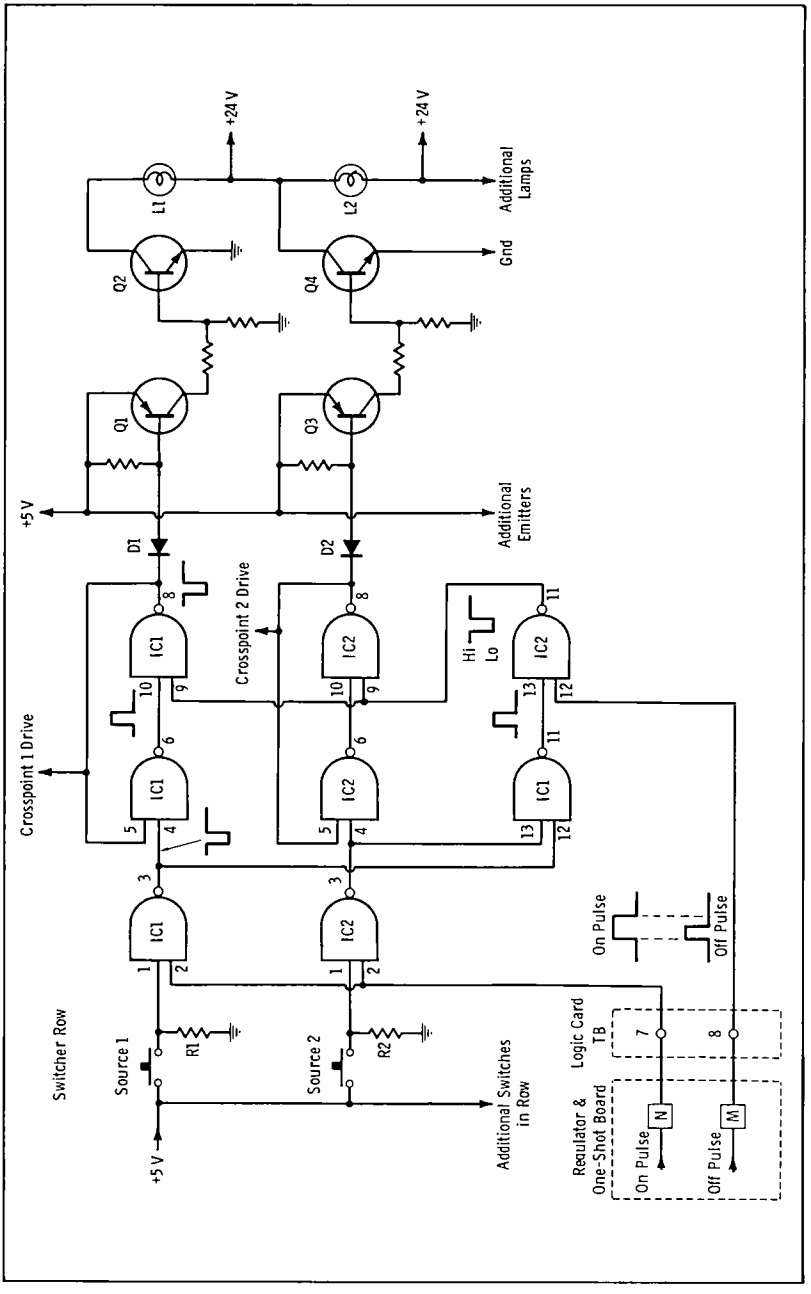


Fig. 9-9. Simplified block diagram of digital-switcher logic card.

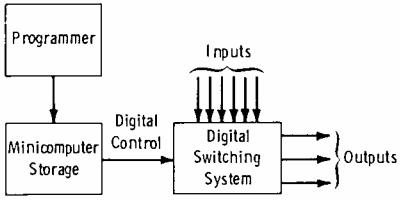


Fig. 9-10. Simplified block diagram of computer-controlled switcher.

sources of upcoming events; permit manual control of on-air-event timing; provide for last-minute changing of time and content of upcoming events; store a full day of programming by use of punched-paper-tape or punched-card input; and provide for integration with data-processing equipment in the traffic, sales, programming, and accounting departments.

An automation switcher[2] is particularly useful for the average station break, commonly termed the "panic period" because of the many operations required over a relatively short time span. The programmed nature of automation switchers eliminates human error, and stations report an average of 95% of switching errors are human errors, made by competent operators.

Fig. 9-10 shows a simplified block diagram of computer-controlled digital swtiching. The upcoming program information is stored in the minicomputer and is read out either by real-time clock pulses or by a fixed sequence of signal-source start-and-stop operation.

9-3. DIGITAL TAPE-TRANSPORT CONTROL

In tape recording, it is frequently desirable to employ a high-speed search unit as a position locator, to guide the tape to a predetermined destination held in memory. Digital techniques lend themselves readily to such applications.

The Oliver Audio Engineering (OAE) Model 400-RSM digital position locator is an example of digital techniques applied to tape-transport control. The basic transport timing elements as mounted in a Scully recorder are shown in Fig. 9-11. The OP-4 optical pickup card uses two precision-aligned infrared sensor pairs (one LED and one phototransistor) to detect reflective lines on a revolving strobe disc. The self-adhesive strobe disc is mounted to the top of the supply-reel brake disc. The OP-4 pickup card is mounted on the bottom of the torque-motor brake plate, directly over the strobe lines on the strobe disc. Optimum clearance is 1/16 inch or less.

2. For a full description of a tv digital automation switcher, see: Harold E. Ennes, *Television Broadcasting: Equipment, Systems, and Operating Fundamentals* (Indianapolis: Howard W. Sams & Co., Inc., 1971) pp 332–341.

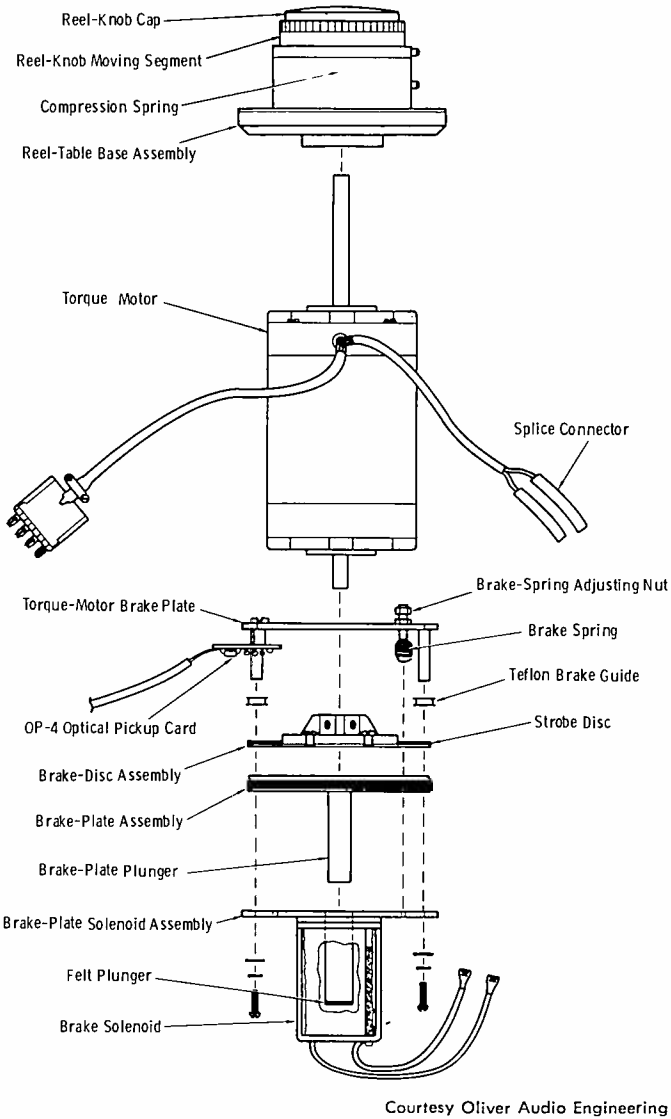
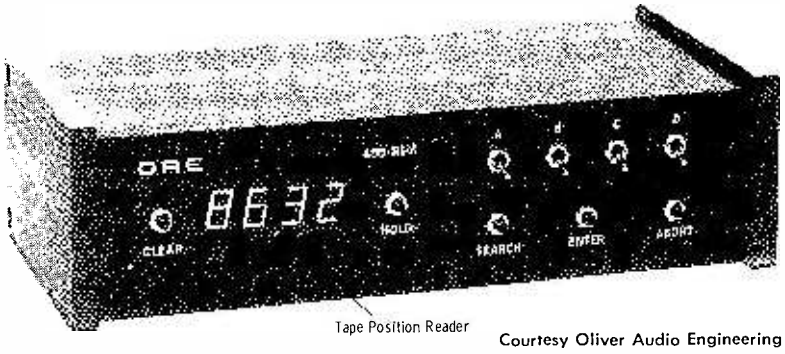


Fig. 9-11. Strobe disc and optical pickup card for tape recorder.

The OP-4 pickup is TTL, two-phase at 100-kHz maximum frequency. The detected signals are amplified, compared with a reference voltage, and transmitted at TTL logic levels to a 400-Series reader or search unit. Two ten-turn trimmers are used to adjust the pickup sensitivity.



Courtesy Oliver Audio Engineering

Fig. 9-12. Control panel of Model 400-RSM digital transport control system.

A front-panel view of the 400-RSM system is shown in Fig. 9-12. The controls illustrated have the following functions:

CLEAR: Resets position reader to zero
HOLD: Disables input to position reader
A, B, C, D: Select memory addresses
ENTER: Transfers tape position to memory
SEARCH: Initiates search process
ABORT: Terminates search process

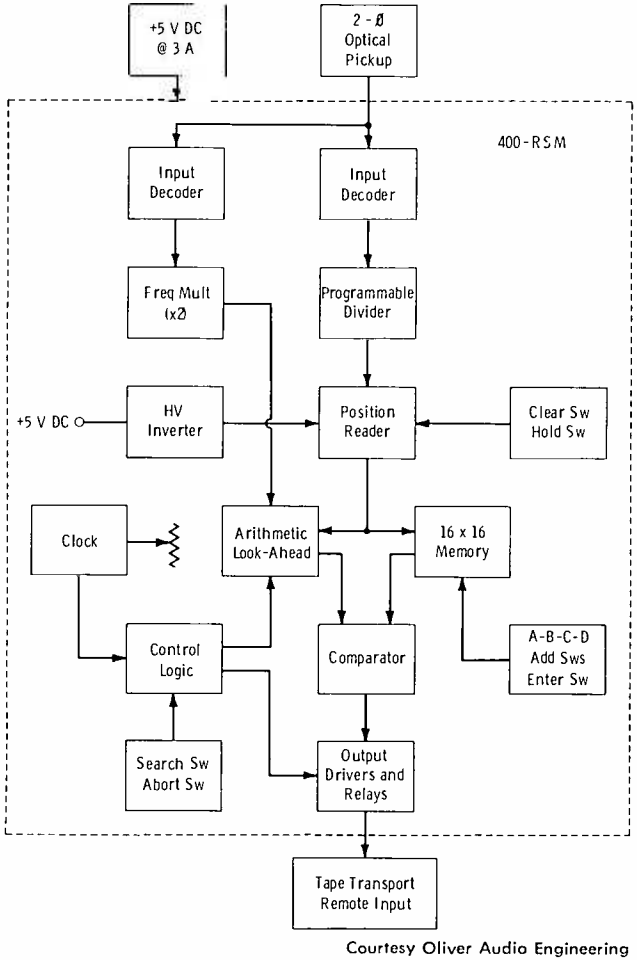
The transport output controls are fast forward, rewind, and stop. All are via spdt Form-C relay contacts.

Operation of this unit is straightforward. Pressing the ENTER button transfers the existing tape position to one of 16 memory locations, determined by the memory-address switches. To return to that position later, the operator sets the memory-address switches and presses SEARCH; the rest is automatic.

The search module compares the tape position with the search location in the memory (Fig. 9-13). The transport is placed in the proper mode (fast forward or rewind), and the tape is accelerated toward the search location. Just prior to reaching the destination, the tape is decelerated down a "digital glide slope" (adjustment on rear of panel) and stopped at the search location. The search module then switches back to standby, and transport control is released to the operator.

The adjustable "digital glide slope" in this unit may be set to correspond with the maximum deceleration rate of a particular transport. This provides search times close to the theoretical limit of the transport without overshoot.

The memory provides 16 four-digit destination locations. The code used is straight binary with binary 0000 representing location 1 and binary 1111 representing location 16.



Courtesy Oliver Audio Engineering

Fig. 9-13. Block diagram of Model 400-RSM.

Reviewing Fig. 9-13, you will observe that a high-voltage inverter is connected to the position reader. This steps a 5-volt source up to the voltage required for the four 0.55-inch gas-discharge displays used in the position reader. The basic power supply is +5 volts dc at 3 amperes.

9-4. PROGRAMMED VIDEO-TAPE EDITING

Recent years have witnessed the progress of tape-splicing techniques from the razor blade to the electronic method. *Programmed*

electronic editing permits a computerized, automatic-programming type of editing operation.

Early (manually controlled) electronic editing removed the necessity for cutting the tape and eliminated the audio-video offset problem. Using devices to count edit pulses provided a means for identifying individual frames. The pulses also provided proper timing for performing push-button electronic edits. This process still left much to be desired, however, and was primarily limited to "assemble" editing functions. The final breakthrough to modern electronic editing came in the mid 1960s with the addition of a serial time code. This code, recorded on the audio cue track, provided a convenient means of frame identification at any point on a given tape.

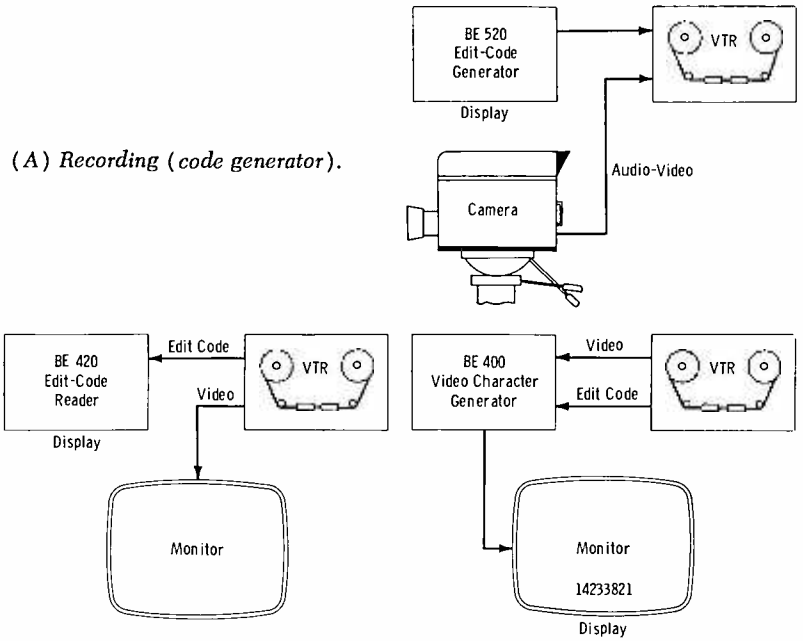
Modern electronic (code-controlled) editing has added several new dimensions to the business of producing material for television. Easy edit-point selection is facilitated through high-speed search of the recorded material. Cueing to a selected edit point is a simple automated function. Tape locations are easily logged, and, with frame-by-frame control, highly sophisticated editing procedures may be employed. Complete preview before edit is easily implemented. Most important, these new systems increase efficiency and reduce costs.

The SMPTE edit code is generally employed in automated tape-editing systems. The reader should review Section 5-6 before proceeding.

Although the specific Electronic Engineering Company (EECO) equipment described in this section includes refinements over some simpler systems, a brief description of it will serve to illustrate fundamentals of all modern programmed editing systems. The basic equipment consists of an electronic editor programmer and a time-code generator. Accessory equipment includes an auxiliary programmer, an auxiliary time display, and a remote transport control panel.

Courtesy Electronic Engineering Co. of California

Fig. 9-14. Video character generator (left) and edit-code reader.



(*A*) *Recording* (*code generator*).

(*B*) *Reviewing programming* (*code reader*).

(*C*) *Reviewing programming* (*character generator*).

Fig. 9-15. Basic applications of indexing-editing equipment.

The EECO "Mini-Module" units are normally used in pairs. Fig. 9-14 shows two of these units. The BE 520 (not illustrated) generates an electronic signal (edit code) that is recorded on video tape. Either the BE 420 or BE 400 (right and left, respectively, in Fig. 9-14) can read this signal as taped video scenes are monitored, and either unit provides a visual display of the edit-code signal expressed in terms of real or elapsed hours, minutes, seconds, and frames. All three units are small (approximately $5 \times 8 \times 11$ inches), hence the term "Mini-Modules." Basic applications are outlined in Fig. 9-15. The units are completely solid-state, using integrated circuits and transistors.

The BE 520 Edit-Code Generator

The BE 520 edit-code generator produces the indexing code needed for efficient editing of video tapes. The code is in the form of an electronic signal recorded on the cue channel or second audio channel, and represents real or elapsed time expressed in hours, minutes, seconds, and frames. Normally, the edit code is recorded

on the tape as scenes are shot, but it can also be added to previously recorded tapes.

Some of the advantages and options available when video or audio tapes carry the edit-code signals are:

1. Tapes can be previewed and a sequence log written to show start and stop times of selected scenes.
2. Subsequent program editing, either manual or computer-automated, can be accomplished more quickly and at less cost than with other methods.
3. Scenes to be shown in a live program can be located more rapidly either manually or automatically.
4. The edit code can be used to synchronize dual-equipment operations.
5. Recordings of significant events carry a permanent time record. Sequences and single frames can be examined in terms of actual time and with split-second accuracy.
6. The edit code can accompany the transfer of video scenes from one type of recorder to another. For example, quadruplex recordings can be copied on a helical recorder. The low-cost equipment can then be used for careful scene reviewing and development of a program sequence log. Subsequent editing of the original quadruplex tapes can be made with minimum use of production-studio time, including automatic or computer-automated editing equipment.
7. Elaborate computer storage, retrieval, and editing systems are already used for some types of video programs. The edit code is used for reference indexing, storage retrieval, and program sequencing.
8. The edit-code index is never dependent on the original tape length and mechanical footage count. Any section of tape carries its original edit-code identification.

The generator utilizes either the power-line frequency or an externally applied video/sync signal as its reference frequency when generating time at frame rates of 25 Hz (European) or 30 Hz (USA), and the time of day coded into the output time code is then as accurate as the selected reference frequency. At the 29.97-Hz color frame rate, however, the time of day will be in error. This error is due to the fact that the generator counts as if the video frame rate were 30 frames per second, or 1800 frames per minute. However, at the rate of 29.97 frames per second, only 1798.2 frames per minute are generated. Therefore, an error of 1.8 frames per minute ($1800 - 1798.2 = 1.8$) is induced. Because the basic video frame duration is 1/30 second, or 33⅓ milliseconds, the time error is 60 milliseconds ($1.8 \times 33\frac{1}{3} = 60$) per minute. To overcome this error, the gen-

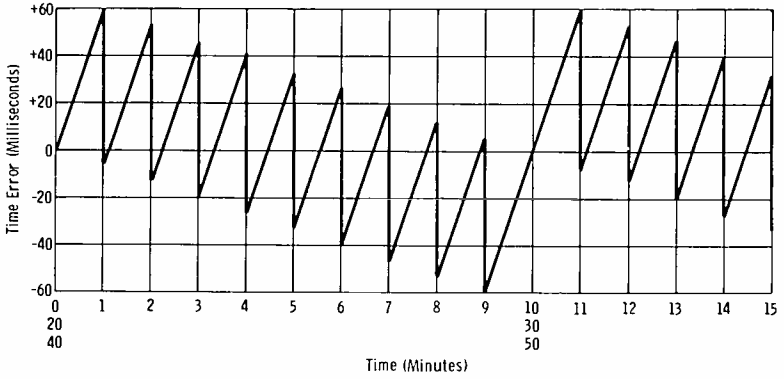


Fig. 9-16. Time-error correction.

erator drops two frame counts per minute for a time correction of 66⅔ milliseconds. However, this effectively overcorrects the error by −6⅔ milliseconds ($60 - 66\frac{2}{3} = -6\frac{2}{3}$). (See Fig. 9-16.) At the end of the second minute, the total error before correction is 53⅓ milliseconds. After the two-frame correction, the total error is −13⅓ milliseconds. The two-frame correction is repeated until after the ninth-minute correction, when the total error is −60 milliseconds. Between the ninth and tenth minutes, the 60-millisecond error generated returns the total time error to 0 milliseconds. Therefore, it is not necessary to make the two-frame correction at the tenth minute.

Since the color video frame rate is nearer 29.97002618 than 29.97, there is actually an accumulated error of about 0.51 millisecond in ten minutes. This amounts to a total error of about 75 milliseconds (0.075 second) per day. If the two-frame error correction were not made each minute (except the tenth), there would be an error of 86 seconds per day.

The generator utilizes either the power-line frequency or an externally applied video/sync signal as its reference frequency. The video/sync signal is applied through either connector J1 or J2 (Fig. 9-17) to the frame sync separator, which separates the sync information from the incoming signal.

The internal power supply operates from either 117 or 234 volts, at 50 to 60 Hz, to provide the dc voltages required by the circuits in the generator. The power supply also generates a 50- or 60-Hz signal (depending on the operating power-line frequency) that is applied to a squaring amplifier. The squared signal is applied to a flip-flop, which generates a signal at half the power-line frequency. The source-switch circuitry selects either the video/sync or the half-power-frequency signal and applies the selected signal to the phase comparator as the reference-source signal.

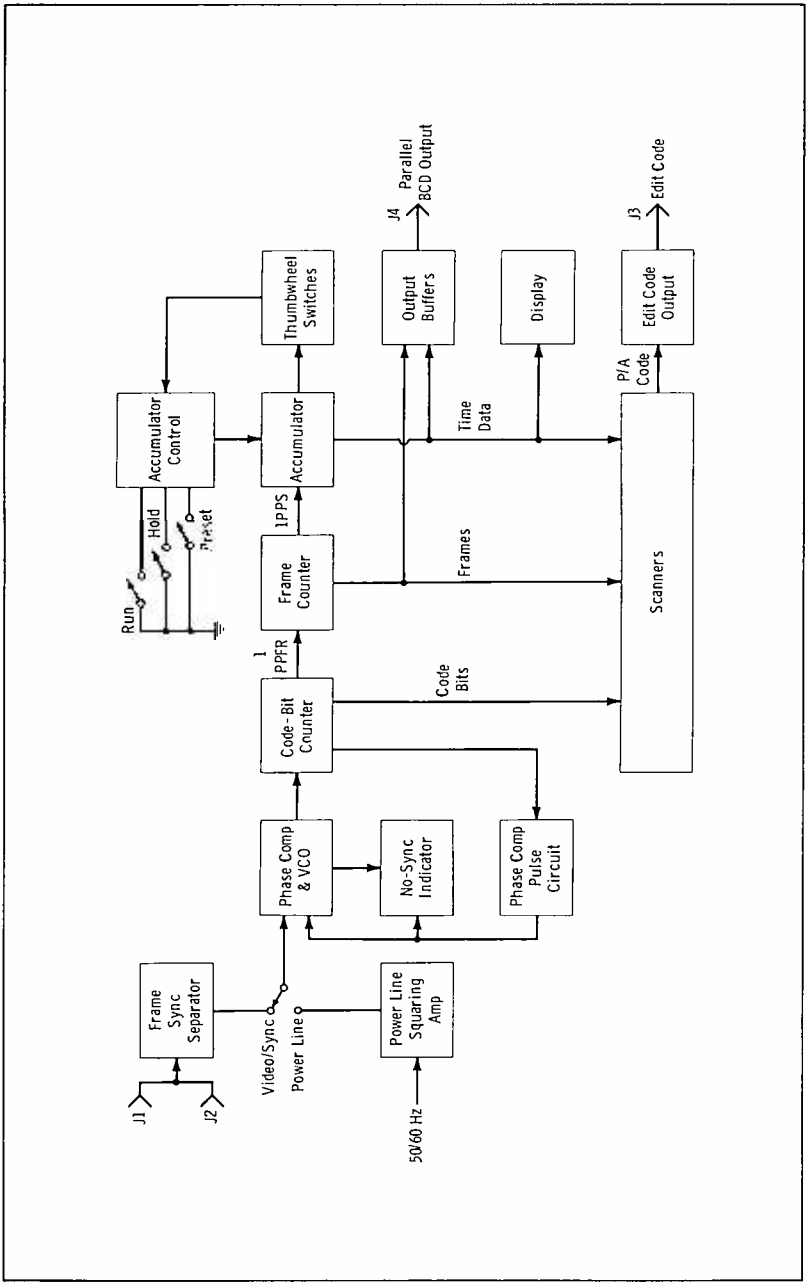


Fig. 9-17. Simplified block diagram of EECO BE 520.

The phase comparator compares the reference-source signal with an output signal from the phase-comparator pulse circuit. The frequency difference, if any, is converted into an error voltage that is applied to the voltage-controlled oscillator (vco). The vco is a free-running oscillator whose basic frequency is changed by the error voltage. A derivative of the vco frequency is fed back to the phase comparator. This forms a phase-locked loop that synchronizes the vco output frequency with the selected reference-source signal. A no-sync indicator circuit gives a visual indication when the vco is not synchronized.

The output frequency of the vco is divided by the code-bit counter, producing various code-bit signals and frame-rate signals. The code-bit signals and clock signals are applied to the scanners to generate the sync-word and frame information for the edit code. A one-pulse-per-frame-rate (1 ppfr) signal is applied to the frame counter to update the frame count. An inverted 64-code-bit ($\overline{64CB}$) signal is also coupled through an output buffer to J4 as a 1-ppfr output signal.

The frame counter generates frame-count information. When the reference frequency is a 30-Hz video/sync signal, a 29-97-Hz color video/sync signal, or a 60-Hz power-line frequency, the selected frame count is 30. When the reference is a 25-Hz video/sync signal or a 50-Hz power-line frequency, the selected frame count is 25.

The outputs of the RUN, HOLD, and PRESET switch circuits are applied to the accumulator control. The control signals reset all the accumulator outputs to zero, preset the accumulator output at a 50-kHz rate to any desired preset time, advance the accumulator at a 1-pps rate, or hold the accumulator from counting.

The accumulator generates the time of day, 0 to 24 hours, at a 1-second rate. This time is constantly available at the accumulator outputs in parallel bcd (1-2-4-8) form. The accumulator-generated time is displayed by numeric indicator tubes. Also, the generated time and the frame count are applied through output buffers to connector J4 as parallel bcd signals.

The scanner receives the time-of-day information from the accumulator, frame information from the frame counter, and signals from the code-bit counter to produce the edit code. The scanner also generates a sync word which is included in the edit code. The edit code is applied to an edit-code output circuit for the necessary amplification, buffering, and impedance matching before being applied to output connector J3.

Space does not permit a detailed description of the circuitry in this equipment. However, a brief analysis of the phase comparator and the vco will assist the reader in becoming acquainted with the type of logic circuitry involved. See Fig. 9-18.

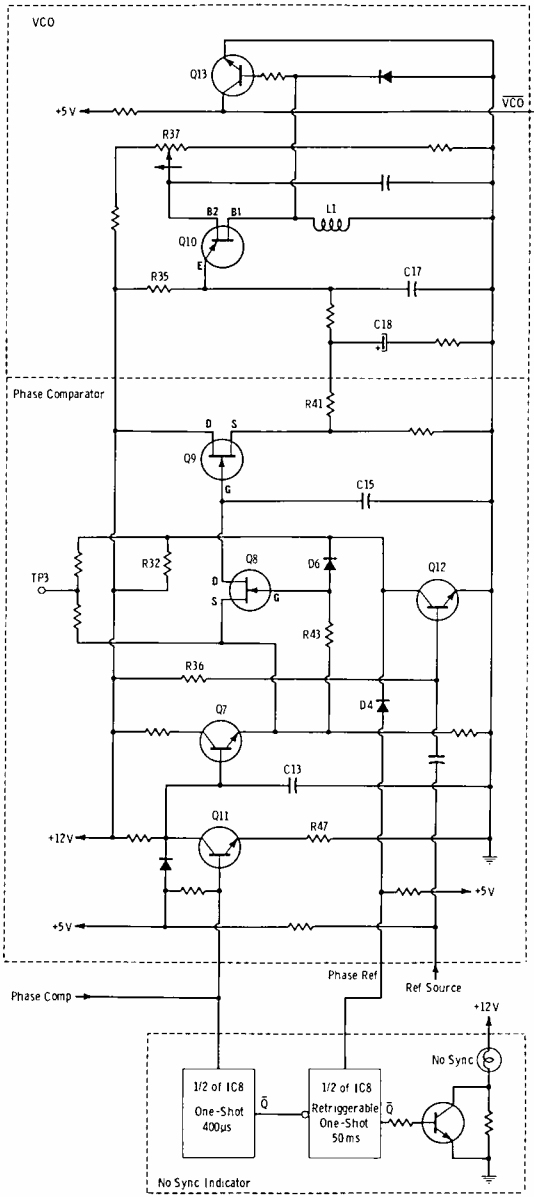
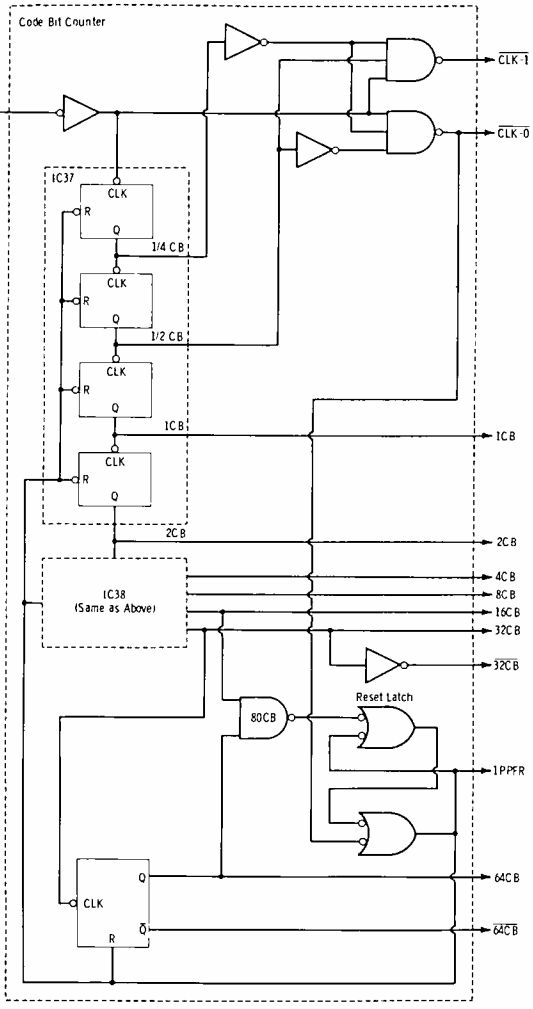


Fig. 9-18. Simplified logic diagram of



phase comparator and vco section of BE 520.

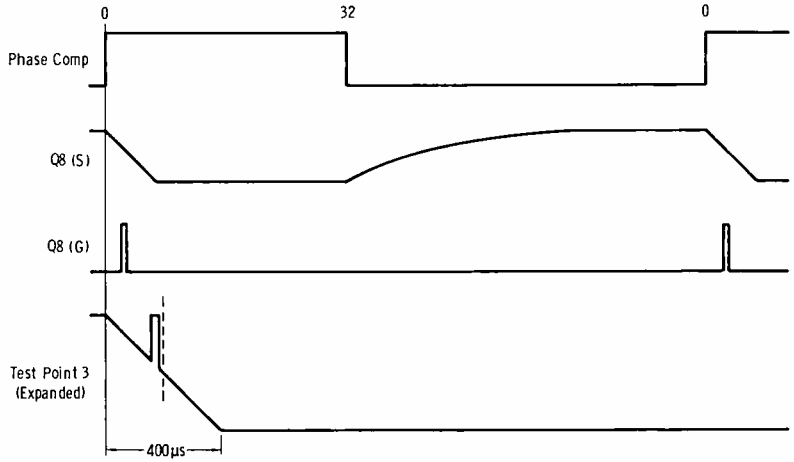


Fig. 9-19. Phase-comparator waveforms.

The voltage-controlled oscillator (vco) generates the basic frequency used throughout the generator to form the code bits and clock pulses. The vco is a free-running relaxation oscillator whose frequency is controllable by the input dc voltage level from the phase comparator. The vco output frequency is divided by circuits in the code-bit counter and then reapplied to the phase comparator at a one-pulse-per-frame rate. Therefore, a phase-locked loop is formed. When the phase-comparison pulses are compared with a one-pulse-per-frame-rate reference-source signal, it is possible for the phase comparator to lock the vco frequency accurately to the incoming video/sync signal or power line.

The phase-comparison signal is a positive-going pulse having a duration of 32 bits and a frequency of one pulse per frame (Fig. 9-19). The positive phase-comparison signal causes Q11 (Fig. 9-18) to conduct at a constant current determined by R47. This constant current causes the potential on C13 to decrease at a linear rate, forming a negative-going linear ramp with a duration of approximately 400 microseconds. This pulse is applied to the source (S) of FET Q8, through emitter follower Q7.

A narrow reference-source pulse, derived from either the video/sync signal or the power line, is applied to the base of transistor Q12. When the reference-source pulse goes low, the collector of Q12 goes high, thereby back-biasing diode D6. With D6 back-biased, there is no current through resistor R43. Therefore, the voltage difference between the source and the gate of Q8 is zero, causing Q8 to conduct. Capacitor C15 then charges to the voltage at the source (S) of Q8. Because Q8 is turned on at the frame rate (25 or 30 per

second), the voltage level at C15 is maintained and applied to the gate of FET Q9. This FET functions as a source follower, so the voltage level established at its gate also appears at its source.

The oscillator portion of the circuit consists of unijunction transistor Q10, capacitor C17, inductor L1, resistor R35, and potentiometer R37. Capacitor C17 charges through R35 to a level that will allow Q10 to fire. Capacitor C17 then discharges through Q10 and L1. When the capacitor is discharged sufficiently, Q10 turns off and C17 begins charging again. Potentiometer R37 determines the firing point of transistor Q10 and thereby adjusts the operating frequency of the oscillator.

Capacitor C17 receives an additional charge through Q9. Therefore, the point on the ramp at which Q8 fires also determines the frequency of the vco. Because a derivative of the vco signal is utilized by the phase comparator, the oscillator is self-compensating and will lock onto the reference-source signal.

Test point 3 provides a convenient point for monitoring the adjustment of the vco frequency. With an oscilloscope, both the reference-source pulse and the phase-comparison-signal ramp can be monitored. Potentiometer R37 is adjusted to position the reference-source pulse to coincide with the ramp (Fig. 9-19).

In addition to the actions described above, a phase-reference pulse is coupled through diode D4 to the no-sync indicator circuit.

The BE 420 Edit-Time-Code Reader

The BE 420 "reads" the standard SMPTE edit code from any source and displays it as real or elapsed hours, minutes, seconds, and frames. This display is available as reference during tape play, rewind, fast forward, or single-frame hold. Scenes from quadruplex or helical tapes may be reviewed and a program sequence developed to show the start and stop times of selected scenes. The program sequence provides an accurate index for manual, automatic, or computer-automated editing of the tapes.

The BE 420 also has helpful supplemental outputs. For example, a field-rate output can be used to keep a multitrack audio recorder in sync with video playback; this eliminates the need for a separate control track in double-system operations (Fig. 9-20). Parallel edit-code outputs are used in computer/automated editing and other equipment controls.

A hold button allows displayed time to be frozen at any point to aid scene identification. Once logged, any scene can be quickly relocated for inclusion in a live broadcast. Significant events can be analyzed in terms of real time.

The incoming SMPTE edit-time code is applied through connector J1 to the demodulator (Fig. 9-21). The demodulator utilizes the

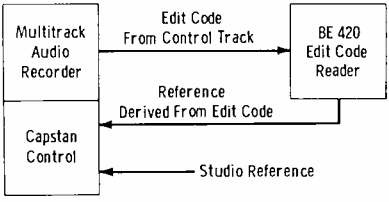


Fig. 9-20. Method of synchronizing audio recorder having control servo.

edit-time code to produce two signals—data ones and code clock. The data-ones signal is a series of positive pulses, each being one-half bit count in duration and each occurring when a one transition occurs in the incoming edit time code. The code clock is a series of 80 positive pulses per frame, each with a period of one code bit and a duration of approximately one microsecond. Each code-clock pulse is timed so that it occurs just prior to the end of a data-ones pulse (if present).

The data-ones and code-clock signals are applied to a decoder. Both signals are inverted (identified as $\overline{\text{CLK}}$ and $\overline{\text{ONES}}$ in Fig. 9-21) and applied to buffer amplifiers for output signals at J2. Two code-bit signals ($\overline{\text{4CB}}$ and $\overline{\text{64CB}}$) are derived from the code clock and applied through buffers to output connector J2. The sync-word portion of the data-ones signal is decoded ($\overline{\text{FRWD}}$) and also applied as an output voltage level at J2 through a buffer. Eight enabling signals (labelled $\overline{\text{1F EN}}$, $\overline{\text{10F EN}}$, $\overline{\text{1S EN}}$, $\overline{\text{10S EN}}$, $\overline{\text{1M EN}}$, $\overline{\text{10M EN}}$, $\overline{\text{1H EN}}$, and $\overline{\text{10H EN}}$ in Fig. 9-21) are produced by the decoder and applied to counter/registers to distinguish properly the various data ones (frames, seconds, minutes, or hours) from each other. The decoder is capable of determining the tape direction (forward or reverse) and, by utilizing a left-right shift register, transferring the decoded serial data (1, 2, 4, 8, and enabling signals) to the counter/register. In addition, field-rate, frame-rate, and read-enable signals are produced, buffered, and made available at output connector J2.

The frames/seconds counter/register utilizes frame and second enabling signals ($\overline{\text{1F EN}}$, $\overline{\text{10F EN}}$, $\overline{\text{1S EN}}$, $\overline{\text{10S EN}}$) and the decoded serial data (1, 2, 4, 8) to produce a set of parallel bcd outputs (labelled BCD Frames and BCD Seconds in Fig. 9-21), which are applied to the frames and seconds display indicators and to buffers for outputs at J2. Other signals used by the frames/seconds counter/register are (as labelled in Fig. 9-21): FWRD, Frame Rate, and Read EN. A switch in the frames/seconds counter/register selects the frame count (25 or 30) to which the bcd frames are permitted to register before being reset. This switch must be set by the operator to correspond to the number of frames per second in the incoming edit-time code.

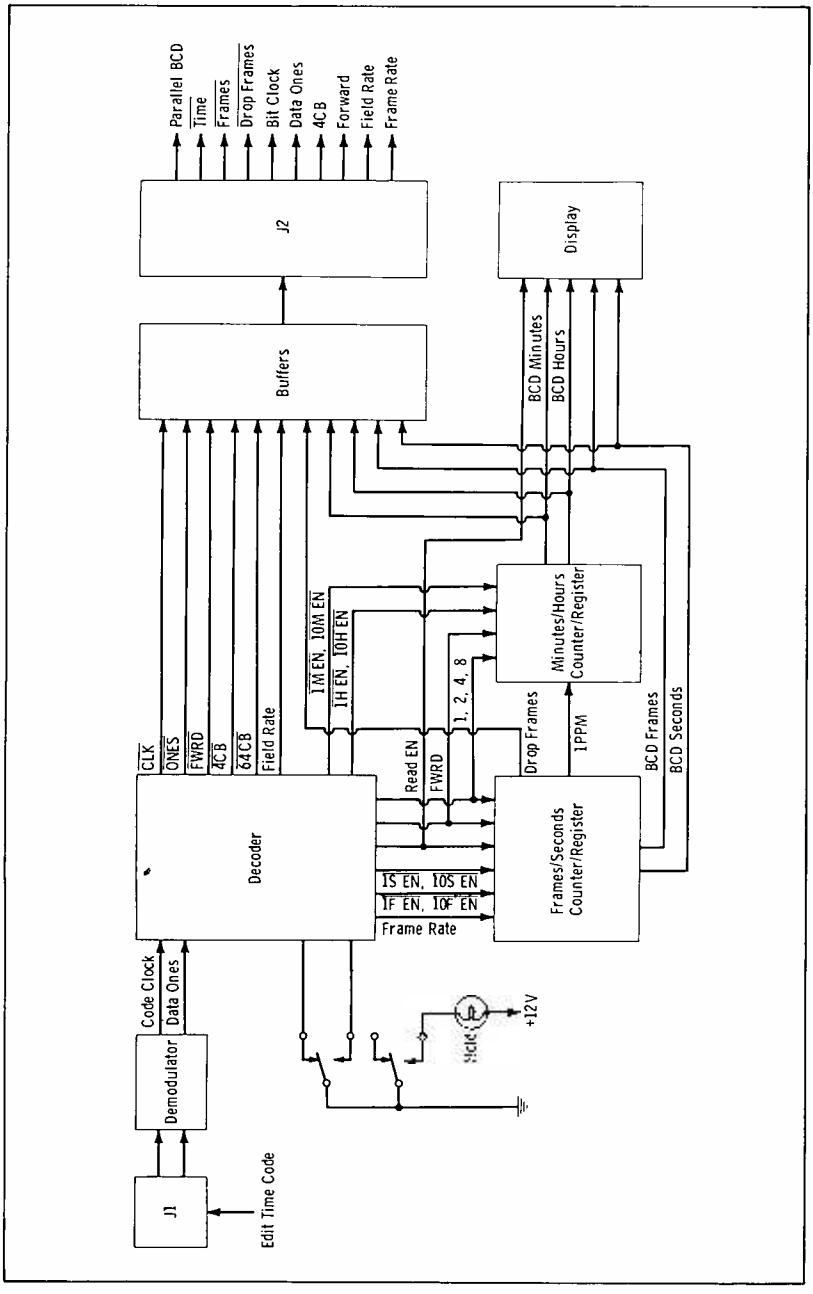


Fig. 9-21. Block diagram of BE 420 code reader.

Similarly, the minutes/hours counter/register produces parallel bcd signals (labelled BCD Minutes and BCD Hours) for application to the minutes and hours display indicators and to buffers for outputs.

The hold switch activates a latch in the decoder that prevents any further decoding of the 1's in the decoder or changing of signals from the counter/register. Therefore, in the hold mode, the display and output signals represent the time and frame count at the instant the hold switch was pressed. When the switch is pressed a second time, the latch is released, and the decoder, counter/registers, and display are permitted to update to the time represented in the edit-time code. The read-enable signal darkens the frames display by disabling the high voltage to the frames indicators.

The BE 400 Video Character Generator

The BE 400 video character generator (Fig. 9-22) converts any number of video monitors into time-code displays. During video play or search, the unit reads the edit code from the tape and generates a corresponding visual time display on the monitor screens. Front-panel switches permit rapid changes in the size and location of the time display, and a black background mask can be added or deleted.

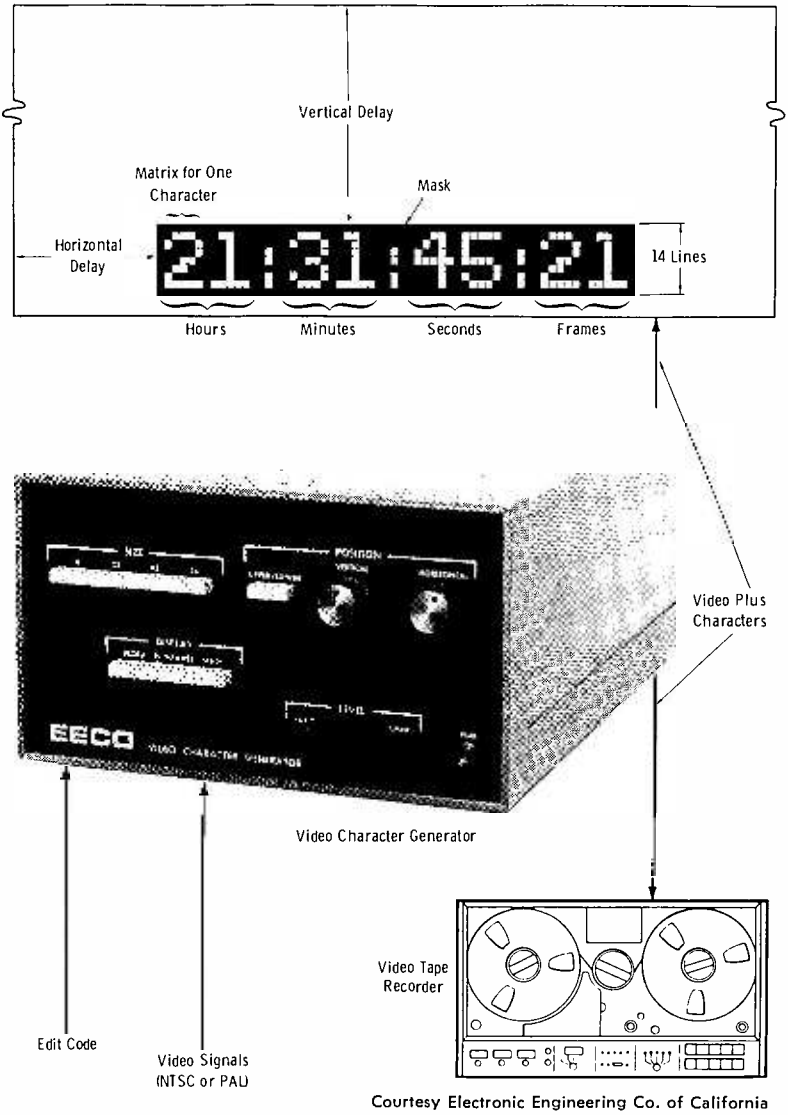
Use of the unit is flexible. It can display the tape edit code (time) on the monitors without affecting the video signals being broadcast or being transferred to another tape. A common option, however, is to add the visual time display to the video signal when it is transferred from a quadruplex recorder to a helical recorder. Scene review is then easy because all scenes, moving or single frame, carry the visual time display.

The features just described make it possible to prepare, with relatively inexpensive equipment, a program sequence log to show the start and stop times of selected scenes. A start-stop time log is useful in a number of operations:

1. Automatic electronic editing
2. Manual electronic editing
3. Computer/automated electronic editing
4. Rapid location of scenes for inclusion in a live program
5. General indexing and filing of taped materials
6. Indexing of computer-stored material

The BE 400 provides a simple method of putting a real-time display on a live video broadcast. It can also provide simultaneous and accurate time references on a number of remote monitors.

This unit accepts either a standard edit code or a parallel time code and frame number input. Control of the output accommodates



Courtesy Electronic Engineering Co. of California

Fig. 9-22. Application of video character generator.

a wide range of viewing requirements. The BE 400 uses nonadditive video mixing to avoid distortion in video processing channels. Inputs and outputs are TTL, RTL, or DTL compatible.

NOTE: Video character generators are more fully described in following Section 9-5.

The BE 210 Electronic Editor Programmer

The BE 210 is the heart of the EECO electronic editing system. It provides precise control of search, cue, and splicing functions. This unit provides transport control based on selection of predetermined start and stop points, and contains the necessary circuitry for programming the recorder edit electronics and recue functions. This circuitry ensures that the edited scene will be the same as the preview.

Features are:

1. Electronic splicing function performed with equal ease for audio, video, or audio and video combined
2. Separate preview and edit functions
3. Automatic frame-by-frame synchronization capability
4. Single-sequence control to automate routine editing operation
5. Push-button touch entry for edit times and control
6. Keyboard entry and/or modification of start and stop times
7. Human-engineered control panel
8. Backlighted control panel for operation in subdued ambient lighting
9. Single-glide recue and search operation

We will go through the basic functions of the BE 210 programmer for the purpose of illustrating how modern electronic editing is computerized. All buttons and switches mentioned are shown in the keyboard drawing of Fig. 9-23.

Modification of Times—Whenever required by the operator, the start and stop times may be recalled and displayed. The start or stop time can be modified by means of a front-panel keyboard assembly. The keyboard is enabled by operating either the + or − button. This operation automatically selects the frame number for modification as displayed on the front panel. If modification of the hours, minutes, or seconds is required, the proper digit is selected by operat-

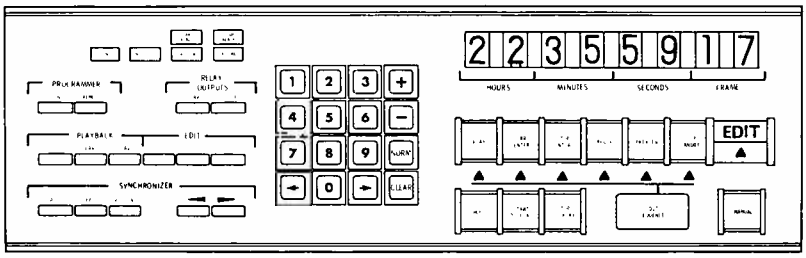


Fig. 9-23. Keyboard of BE 210 programmer.

ing the left arrow button. The proper digit is indicated by two periods located just below the digit.

Keyboard Time Entry—If times are to be entered from a log, the display register is cleared by operating the CLEAR button. This operation resets the display to zero and places the locating periods below the tens-of-hours digit. As the times are entered from the hours to the frames, the indicator automatically advances to the next digit to the right. Operating the NORM button returns the BE 210 to normal operation, displaying the input code.

Sequence Control—In addition to the normally manual operating controls, an edit sequence control is provided for routine edit operation. The normal edit sequence consists of play, start enter (time), stop enter (time), recue, preview, and stop.

Each time the EDIT SEQUENCE button is pressed, the function indicated by a lighted arrow will be initiated. The arrow beneath the next function in the sequence then lights to indicate to the operator the next event in the series. After the stop function has been initiated, the next sequence is recue, preview, stop. This sequence repeats until the edit button is operated. The edit will then be recorded as previewed. The sequence may be interrupted at any point by pressing the appropriate function selector.

Limited-Range Synchronizer—The purpose of the limited-range synchronizer is to provide exact frame synchronization of the associated transport to a reference time code. This capability can be used in transfer editing to synchronize the playback transport accurately with the record transport by referencing to the time code from the record machine.

The limited-range synchronizer has an offset mode in which it will lock the stop-time frame number from its own code to the start-time frame number from the reference code. That is, the synchronizer will phase the playback of its own code in such a way that when the code frame coinciding with its start-time frame occurs, the reference code will be at a code frame coinciding with the stop-time frame. The two codes will then be locked in an offset fashion with the offset determined by the respective stop and start times. This ensures that the start of action of the playback-machine tape is timed to coincide with the ingoing edit point on the record-machine tape.

The limited-range synchronizer also has a nonoffset mode that disregards the start- and stop-time frames and locks frame 1 of the controlled code to frame 1 of the reference code.

Controls are also provided to advance or retard the capstan servo control circuit manually.

Other Controls—Additional front-panel controls provide the following functions:

PROGRAMMER ON: Activates control functions of the BE 210 to control associated video recorder.

PROGRAMMER REMOTE: Delegates control of BE 210 to remote point.

PLAYBACK/EDIT: Six mechanically interlocked buttons.

PLAYBACK CLOCK: The code at the recorder cue-track input is selected. The time-code generator (clock) is normally fed to this point. This section allows the transport (play and stop) to be controlled from the time-code-generator time.

PLAYBACK SEARCH: The video-recorder cue-track output is selected. The code at this point is one of two. During standby or record, the code at the cue-track input appears at the cue-track output through the electronics-to-electronics (E-E) circuit. When the cue track is in playback, the code is reproduced from the tape. This position is normal for tape search and cue. In this mode, the stop time is enabled during recue. The tape may be searched and stopped or cued to any point. A clock or slave code may then be used to place the recorder into play (start). When the recorder is recued, the tape will be rewound to a point (normally 10 seconds) ahead of the stop time.

PLAYBACK SLAVE: This external input is used primarily for transport play control and is also the input from the reference machine for the synchronizer. The second input to the synchronizer is from the cue-track output of the local machine.

PROGRAMMER REMOTE: This button overrides any selected playback input. When the BE 210 is placed in remote, the code is selected remotely. When the clock or slave inputs are selected and the recorder is placed into recue, rewind, or fast forward, the code input automatically switches to the VR code input. This is required because the playback code is needed for these operations.

EDIT A/V: Audio-video edit selection. Audio and video edits are made simultaneously. The edit is begun at the start time (ingoing edit) and terminated at the stop time (outgoing edit). In any of the three edit functions, the edit sequencer is operative. When the recorder is recued, the tape is rewound to a point (normally 15 seconds) ahead of the start time. A recue time longer than that of a playback machine is required to allow code to be read and a play command to be given to the playback machine at the 10-second preroll time ahead of the edit start time. The A/V function is used to start and stop the preview or record of an audio-video edit. The code from the cue-track playback is used in the edit functions.

EDIT A: Audio-only edits are made. See EDIT A/V above.

EDIT V: Video-only edits are made. See EDIT A/V above.

MANUAL: Operation of the manual edit button allows an ingoing edit to be made using the edit button (after play) and an outgoing

edit to be made by operating either play, recue, preview, stop, rewind, or fast forward. Operation of the MANUAL button also resets the edit sequencer to the off condition (no arrows lighted, edit sequence button dim).

START DISPLAY and STOP DISPLAY: Operation of the START (STOP) DISPLAY button transfers the time in the start (stop) storage register into the display logic and displays this time. The time in the store is not altered. This allows stored times to be read and modified. The display is then restored to normal operation by one of three methods. Operation of either the START ENTER, STOP ENTER, or NORM button transfers the display back to the code input selector.

NOTE: The time in one storage register may be moved to another storage register without destroying the time in the first. The NORM button transfers the display to normal operation.

HOLD: The HOLD button freezes the display at the instant of operation. This feature is useful in displaying time without time entry. The display is restored to normal operation with the NORM button. The displayed time may be entered into the start or stop store if required. Either entry button restores the display to normal.

Status Indicators—Indicators are provided on the upper left section of the BE 210.

SYNC: Lights to show that the synchronizer has synchronized local machine to remote code.

NO EDIT: Indicates edit was inhibited.

PREVIEW: Indicates that the video and/or audio monitors have switched to the incoming program.

RECORD: Indicates that the video and/or audio monitors have switched to the incoming program and that the machine is recording.

START READY: Indicates that the start-time function is enabled.

9-5. DIGITAL CHARACTER GENERATORS

The video character generator receives coded signals (usually 6- or 7-bit codes), produces the appropriate horizontal and vertical deflection signals for a video monitor, and generates brightening signals to correspond with the character represented by the digital code. The basic principles of operation were discussed in Section 5-5, and this section should be reviewed before you proceed.

See Fig. 9-24 for the basic action of a 7×5 matrix dot character presentation on a video monitor. A 7×5 character makeup means there are 7 elements vertically (y elements) and 5 elements horizontally (x elements) for each character, where each "element" is a "dot" formed by a brightening pulse. Except for these elements, the

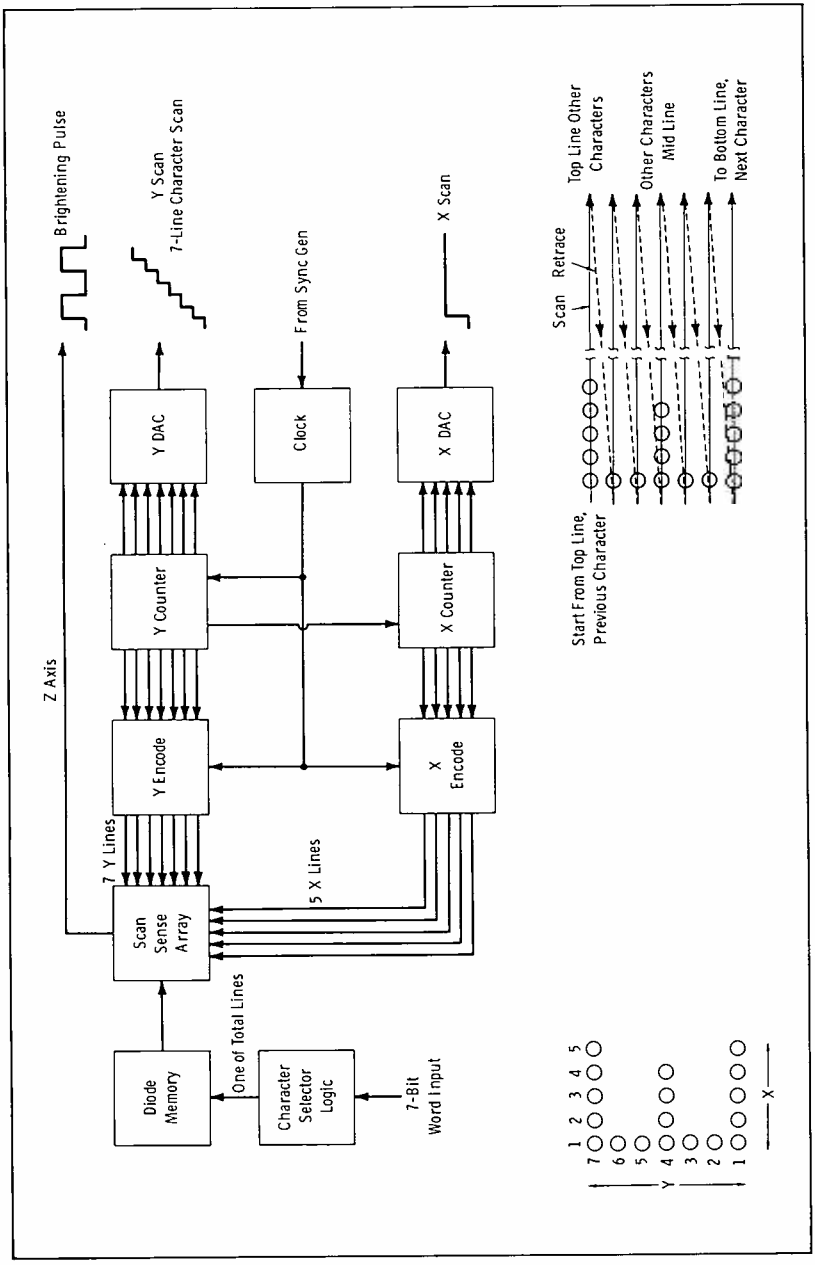


Fig. 9-24. Basic principle of 7 × 5 matrix dot character presentation as generated on a video monitor.

output of the character generator is "in black." However, it is possible for the characters to be "keyed-in" over video from any other video source.

In the example of Fig. 9-24, the scanning beam is stepped through $(7)(5) = 35$ positions that make up a 7×5 dot matrix rectangle. When a character line is activated, the appropriate diodes in the diode memory conduct. When the inputs from the y (vertical) and x (horizontal) encoders coincide with this information, a brightening pulse turns on the z axis. The y and x dac's provide the appropriate voltages to step through the 7×5 dot matrix. In effect, this produces a small raster scan for each character.

A complete row of characters is built up together in horizontal segments. The vertical (y) count is advanced once per character scan line, rather than seven times for each individual character.

The Datavision Model D-3000 Character Generator

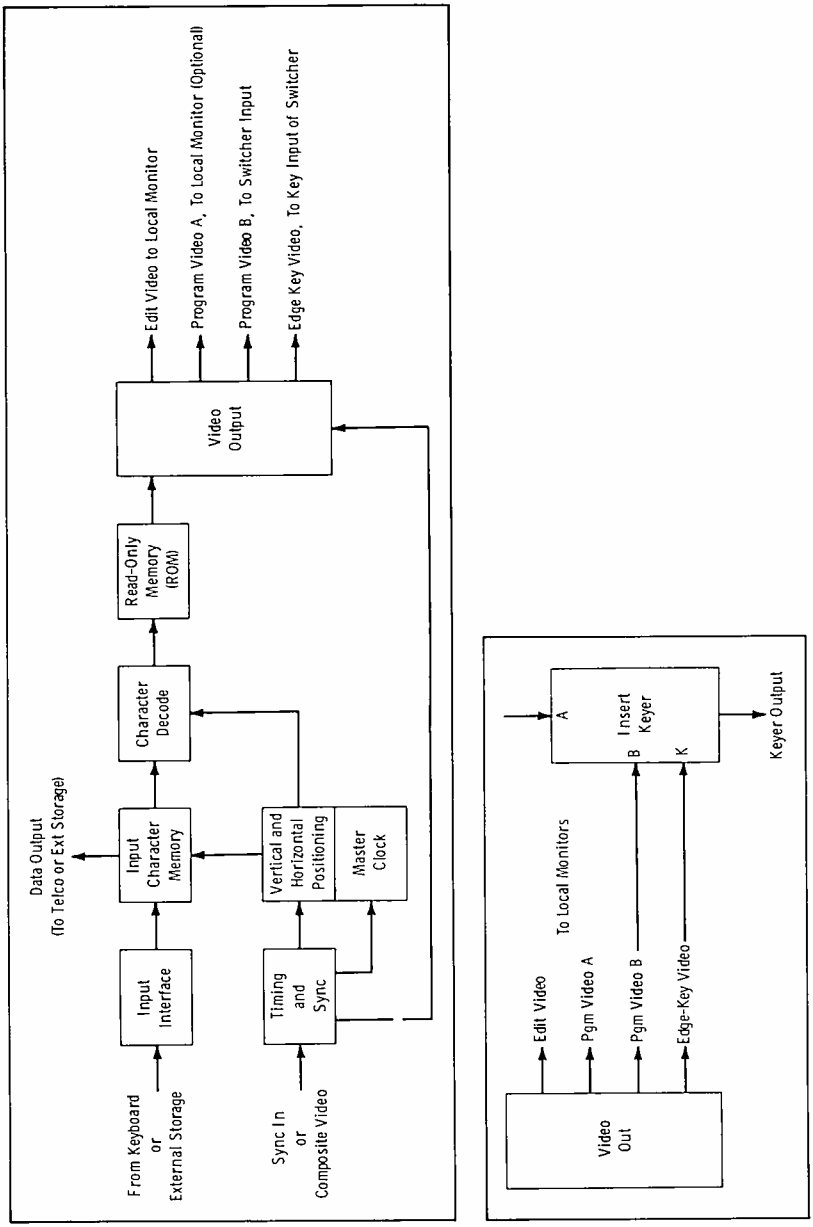
Fig. 9-25A is a simplified block diagram of the 3M-Datavision Model D-3000 television character generator. The Model D-3000 can be used for stand-alone operation, or it can be used to supply outputs for processing in a video switcher. Fig. 9-25B shows the video output details for downstream edge keying. The basic blocks in Fig. 9-25A have the following functions:

Input Interface—The input interface accepts the 8-bit ASCII (modified) code from the keyboard or external storage device. This block acts as a buffer between the input data and the memory, which is a short-term storage location.

The coding used in the 3M-Datavision system is an 8-bit data code. Bits 1 through 7 are the ASCII code (review Section 5-5). Bit 8 is a size control, logic 0 for 28-line characters and logic 1 for 20-line characters (Fig. 9-26). Positive logic is used (logic 1 = high positive level).

Clock (Timing) and Positioning—Timing usually begins with detection of the vertical-sync pulse from an external sync source (or stripped sync from a composite video signal), which synchronizes the internal master clock running at a much higher frequency. This clock develops a series of pulses per horizontal character scan line equal to the resolution capability of the generator.

For example, a character with only a 7-high by 5-wide dot matrix (one dot per pulse) has a total resolution of only 35 video elements, or dots, per character. This is rather coarse resolution. A character formed within an 11-high by 9-wide dot matrix has a total resolution capability of 99 video elements per character, a decided improvement in resolution. Remember in this connection that an 11-high dot structure becomes 22 scan lines in height due to the interlaced scanning of two fields at the 30-Hz scan rate. Through logic-



(A) *Character generator.* (B) *Downstream edge keying.*

Fig. 9-25. Simplified block diagram of Datavision Model D-3000 television character generator.

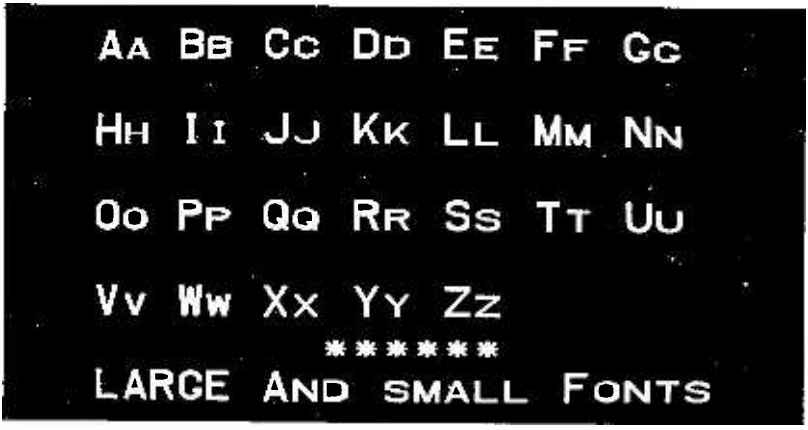


Fig. 9-26. Television display with characters of 20 and 28 scanned lines.

circuit techniques, the size can be "blown up" to any height desired, but with resulting loss of resolution.

Thus, it may be noted that the frequency of the master clock is dependent on the required resolution of the particular system. High resolution is obtained in the Datavision system by a master-clock frequency of about 24 MHz; this is the basic dot rate, which is gated by sync and delivered to the high-resolution character shift registers. The master-clock frequency is divided by three to 8 MHz, which is the basic dot frequency of the low-resolution character set (edit video output of Fig. 9-25). The 8 MHz is then divided by 16 to 500 kHz, and separate 2-μs interval clocks are developed. The character time is represented by the 2-μs interval.

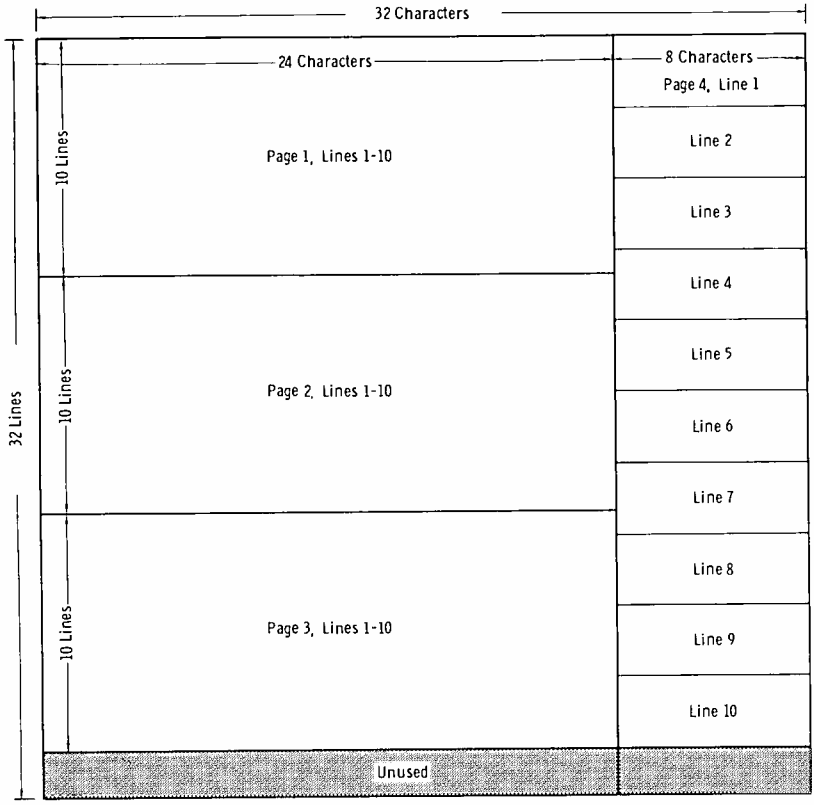
Thus, considering 48 μs out of the nominal 63.5-μs scan interval as the usable picture area, and allowing margins and character spacing, 22 characters can occur on each line.

Positioning allows setting of the left and right margins and the top and bottom lines in the tv raster. A countdown from the low-frequency clock develops 7.5 and 1.875 Hz for use in key repeat and flash circuits.

Input Character Memory—This memory is a read/write memory; information can be stored in (read into) the memory, and the same information can be transferred from (read out of) the memory. New data can replace whatever is currently stored. This section provides storage space equal to the total number of characters that may be displayed on the television screen at one time. Each memory position, or slot, may be related to a position on the television display, and memory loading/unloading timing is derived from the master clock.

The four-page internal memory of the D-3000 is composed of eight 1024-bit integrated-circuit RAMs (Section 6-2) segmented into 40 lines of 24 characters each by means of bcd-to-binary conversion. If each RAM is considered to represent a matrix of 32 characters by 32 lines, the address-conversion circuit yields the format shown in Fig. 9-27A. Fig. 9-27B shows the detailed format. Each page consists of 10 lines of characters with 22 characters per line.

Character Decode and Generation—Information stored in the character memory is still in the data format initiated from the keyboard or external data storage. The character-decoding logic serves to interface between the main character memory and the read-only memory (ROM), interpreting which character in memory is being accessed for display, and directing the ROM to produce the "digital video" pulses required to develop the character on the television raster.

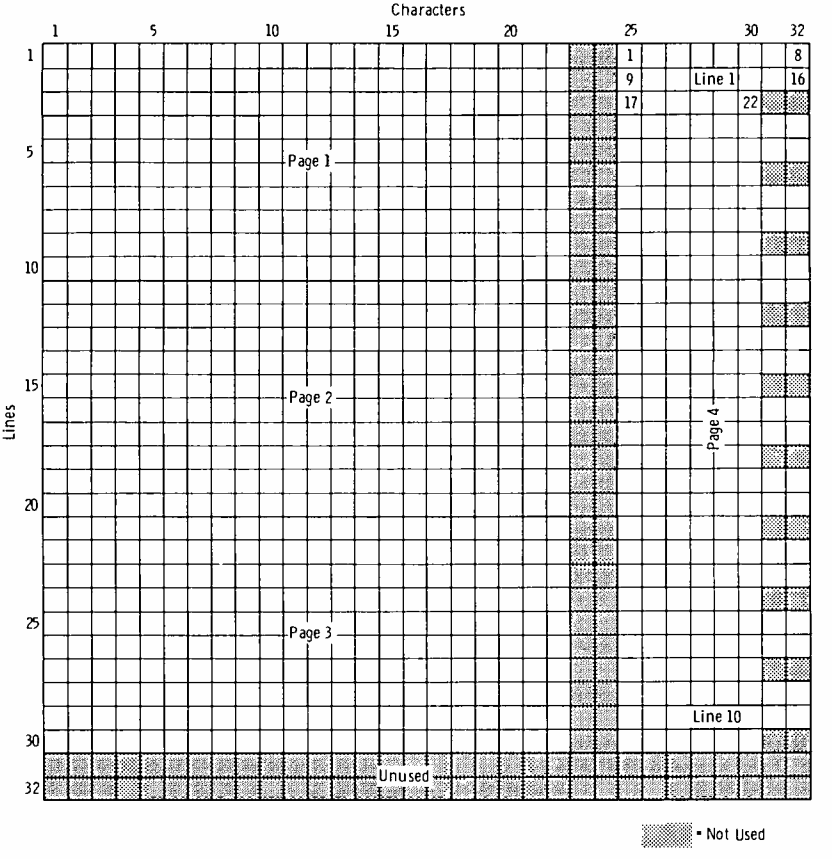


(A) General.

Fig. 9-27. Memory format

The active display area of each line of characters begins four spaces (approx 8 μs) after horizontal sync and continues for 22 character spaces (approx 44 μs). Sixteen scan lines in each field make up the full active character height; full-frame (interlaced fields) character height is therefore 32 lines. All 32 lines make up a large-size edit-channel character; 24 lines make up the small size. On the program channel, the top and bottom pairs of lines are reserved for edge-key generation; thus, the high-resolution characters are 28 lines high for large, and 20 lines for small.

The edit-channel character memory is contained entirely in a single 4096-bit ROM. The code for each character in a given line is presented successively from the main memory. Four additional in-



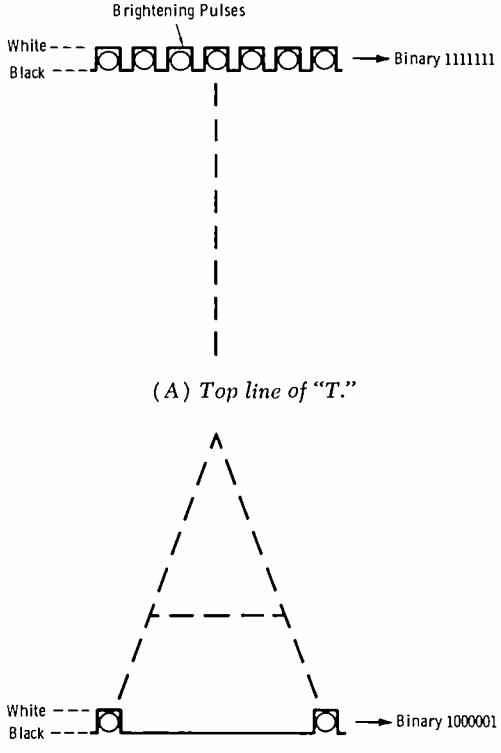
(*B*) *Detailed.*

Courtesy Mincom Div., 3M Co.

of Datavision D-3000.

puts describe which of 16 horizontal segments of the character is to be read. The output of the edit ROM is seven parallel signals representing dots in the character matrix. For example, the top line of a *T* is 1111111, and the bottom line of an *A* is 1000001 (Fig. 9-28). These signals are loaded into a shift register and shifted out at a 4-MHz rate. Seven dots form one row of a character, and an eighth clock pulse adds the space between characters. The edit signal (which feeds the keyboard monitor) is mixed with the visible cursor to indicate the character location.

The program-channel character memory has a much higher bit density and therefore a higher resolution on angles and curves. Each character is 40 dots wide, with eight dots between. The dots are much shorter, resulting in the same character width as in the edit channel. Each of five paralleled ROMs contributes eight of the 40 dots each time a character code is presented to them. The dots are loaded into five shift registers and shifted out at 24 MHz.



(*A*) *Top line of "T."*

(*B*) *Bottom line of "A."*

Fig. 9-28. Examples of dot structure for 9 × 7 matrix.

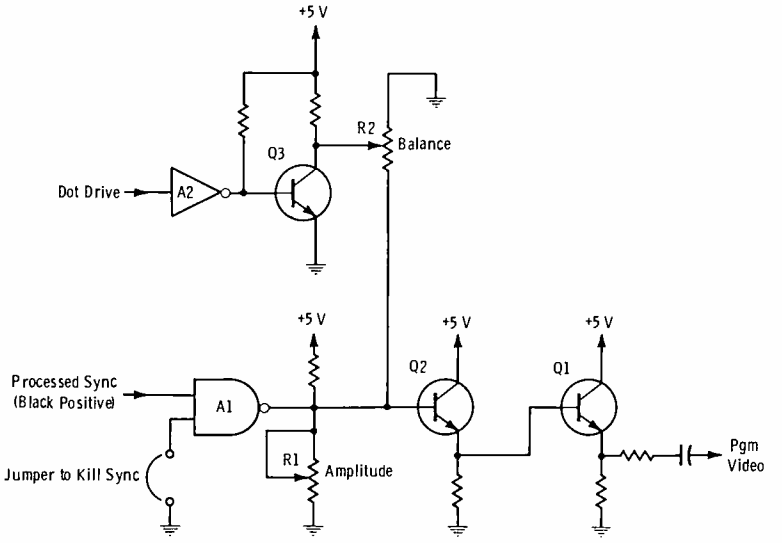


Fig. 9-29. Simplified schematic of single video output in Datavision Model D-3000.

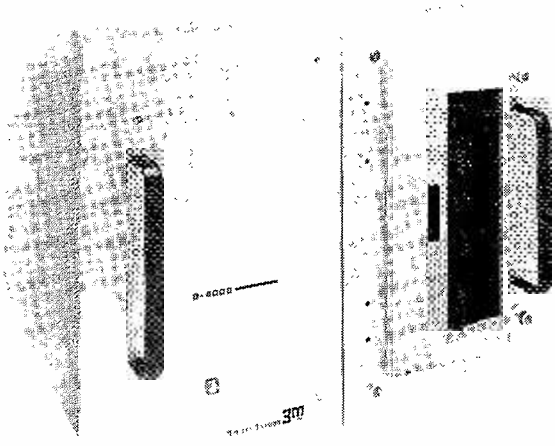
Video Output—It is now necessary to process the ROM output to mix character and edge-key drives with sync to form a standard television composite video signal. Fig. 9-29 shows how this is accomplished in the D-3000. Dot drives are positive voltage excursions at TTL levels. Dot drive is buffered and inverted in A2 and reinverted in Q3 to yield a positive voltage excursion at R2. This is fed to the base of Q2.

Processed sync (positive-going) is buffered by A1 (NAND gate) and presented as a more negative excursion at the base of Q2. The overall level of the mixed signal is determined by R1. The proper sync/video ratio is set by R2. Controls R1 and R2 are adjusted to obtain 0.3 volt sync and 0.7 volt video for a composite signal, when the output is terminated in a 75-ohm load.

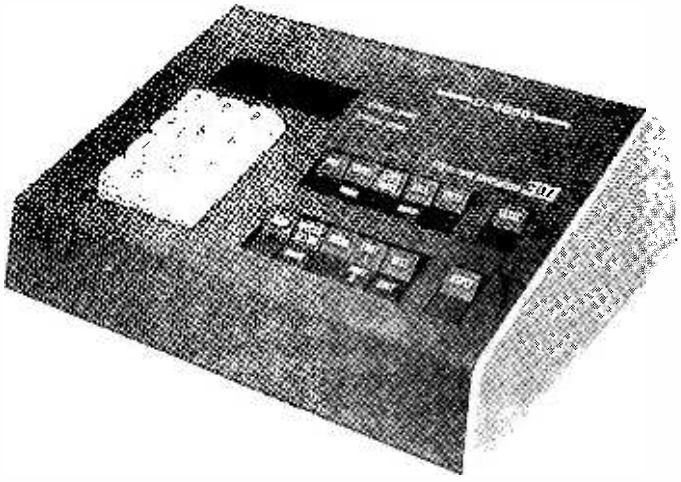
Note that the program channel can be strapped at A1 for noncomposite (sync-free) output. This is done when sync is mixed after the system switcher output.

The Datavision D-4000 External Storage

It will be noted from Fig. 9-25 that the input character memory is capable of feeding an external line or storage facility. This provides the ability to store additional information in an external memory system. Operationally, this external memory capability expands the usefulness of the character generator by providing the means to precompose, edit, store, and retrieve more information (titles, an-



(*A*) *Disc memory.*



(*B*) *Address keyboard.*

Courtesy Mincom Div., 3M Co.

Fig. 9-30. Random-access disc memory and associated keyboard.

nouncements, lists, scores, etc.) than can be held within the internal memory of the character generator.

The Datavision D-4000 random-access disc memory (Fig. 9-30A) utilizes a "floppy disc" for storage. The disc is easily removable, and any number of discs can be used for storage of different infor-

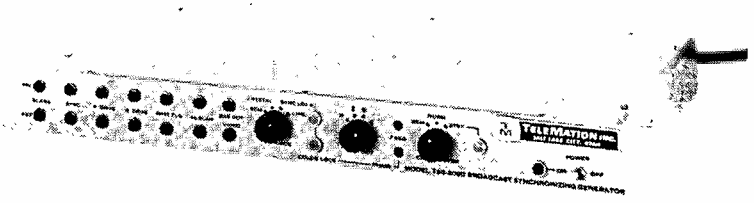
mation. Each disc can store up to 1000 separate, full pages of display. The average search time for any page is less than 0.4 second.

Storage or retrieval is on a page basis to simplify operation. Each address position can store up to 10 rows of information, which means the system actually stores as many as 10,000 rows. Addressing by page allows storage and retrieval of a title in the exact way it was prepared and positioned by the operator during composition.

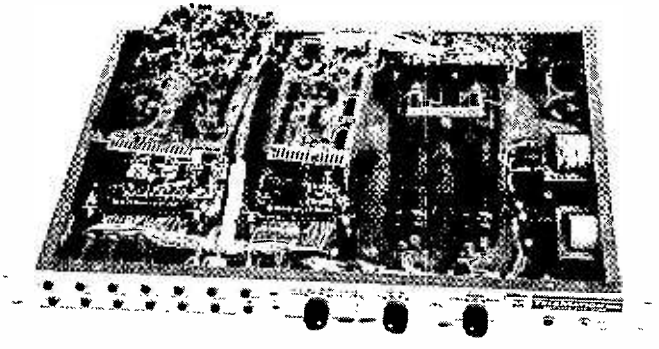
Each D-4000 is furnished with a companion D-4050 address keyboard (Fig. 9-30B) to control the system. This is separate from the disc memory and can be remotely located.

9-6. DIGITAL SYNC GENERATORS

Sync generators employing digital circuitry are becoming commonplace in television broadcast installations. One example is the TeleMation Model TSG-2000 generator. Fig. 9-31A is a front view of this unit, and Fig. 9-31B is an interior view with the various IC cards exposed. The circuitry is all digital, with all waveform transitions (both leading and trailing) derived from master-clock pulses.



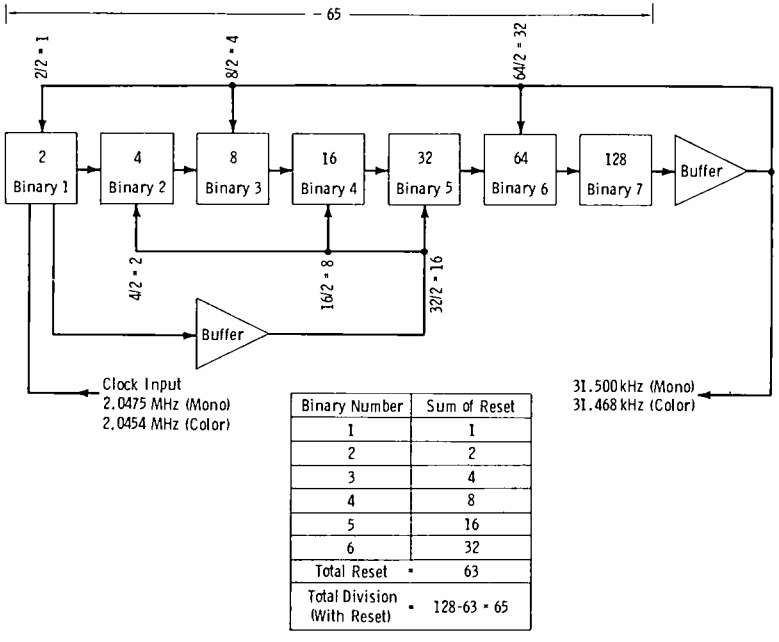
(*A*) *Front View.*



(*B*) *Interior view.*

Courtesy TeleMation, Inc.

Fig. 9-31. Digital synchronizing generator.



Binary Number	Sum of Reset
1	1
2	2
3	4
4	8
5	16
6	32
Total Reset	63
Total Division (With Reset)	128-63 = 65

Fig. 9-32. Divide-by-65 counter system used in TeleMation Model TSG-2000 synchronizing generator.

Integrated circuits are used for all counting and logic functions, and transistors are used in the clock and output circuits.

All digital sync generators have many features in common, with slight variations in application. One feature we must become familiar with is the use of the binary scaler (which normally divides by multiples of two) as an uneven counter giving divisions such as 3, 5, 7, etc. (Review Section 4-6 for principles of operation.)

An example of the type of variation we will find in practice is illustrated in Fig. 9-32. Although two reset paths are employed by TeleMation, the basic principle still applies. The seven identical binary stages have an inherent count of $2^7 = 128$. However, binaries 1 through 6 are reset so that the resultant count is 65.

An easy way to visualize this is shown in Fig. 9-32. For example, the reset pulse into binary 6 is $2^5 = 32$. This is the same as dividing the total count of binary 6 (64) by 2. The table in Fig. 9-32 shows that the sum of the feedback is 63. Thus, a count of 63 is preset into the counter. The total count, 128, minus the preset, 63, results in the desired 65 division of the input clock frequency.

Fig. 9-33 illustrates the color standard for the same sync generator (TeleMation Model TSG-2000). The oscillator is a temperature-

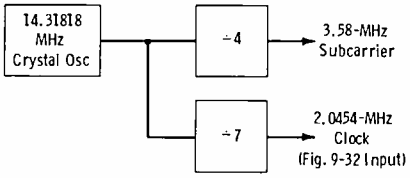


Fig. 9-33. Color standard for Tele-Mation Model TSG-2000 sync generator.

stable, crystal-controlled 14.31818-MHz circuit. This frequency is divided by four and filtered to produce the 3.58-MHz subcarrier output. In addition, the crystal-oscillator frequency is divided by seven to provide the 2.0454-MHz clocking input to the portion of the horizontal-sync logic board represented by Fig. 9-32. The nominal 31.5-kHz signal is divided successively by 3, 7, and 25. This produces the field rate for timing and gating in the vertical-logic section.

All other transitions bearing specific phase relationships to the basic pulse trains are generated in the frequency-dividing process of the counters. These signals are furnished at various points of control within the logic circuitry. By this method, it is possible to initiate, terminate, and gate the pulses of all sync-generator output waveforms directly according to clock timing.

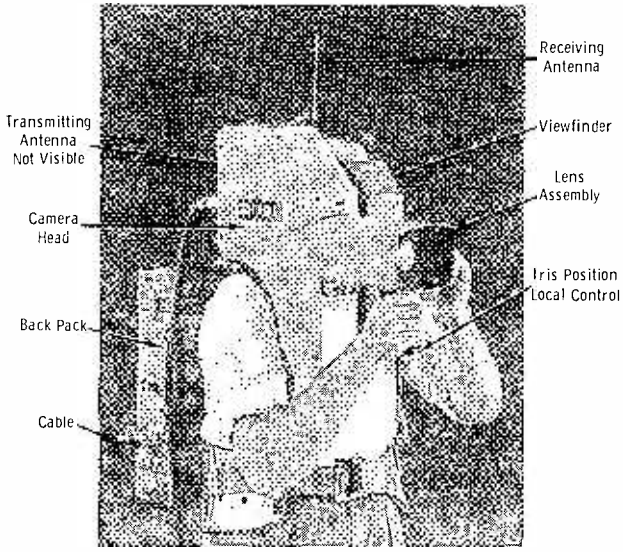
9-7. DIGITALLY CONTROLLED COLOR CAMERAS

Digitally controlled television color cameras that can be interconnected with the camera control unit (CCU) either by rf link (Fig. 9-34) or by a single triaxial cable (Fig. 9-35) have been developed to allow extreme flexibility in operation.

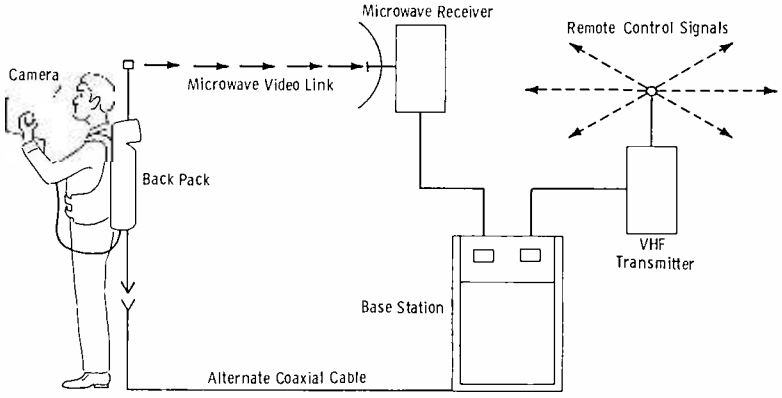
Philips PCP-90 Portable Camera

A portable 3-Plumbicon color camera employing digital control (Fig. 9-34A) was developed by CBS Laboratories, and, under agreement with CBS, it was manufactured and marketed worldwide by Philips as the PCP-90 "Minicam." This camera, with backpack, produces an encoded signal intended for direct broadcasting. All signal processing is accomplished in the backpack so that separate red, blue, and green signals need not be sent to the base station. This reduces the possibility of noise pickup and cuts down on color errors caused by multipath effects.

The camera transmits signals to its base station (Fig. 9-34B) on the microwave frequency of 2 GHz (7 GHz and 13 GHz are optional) from an omnidirectional antenna on the backpack. Remote-control signals to the camera are carried on a frequency of 950 MHz. The command system permits radio control of all functions



(*A*) *Camera and backpack.*



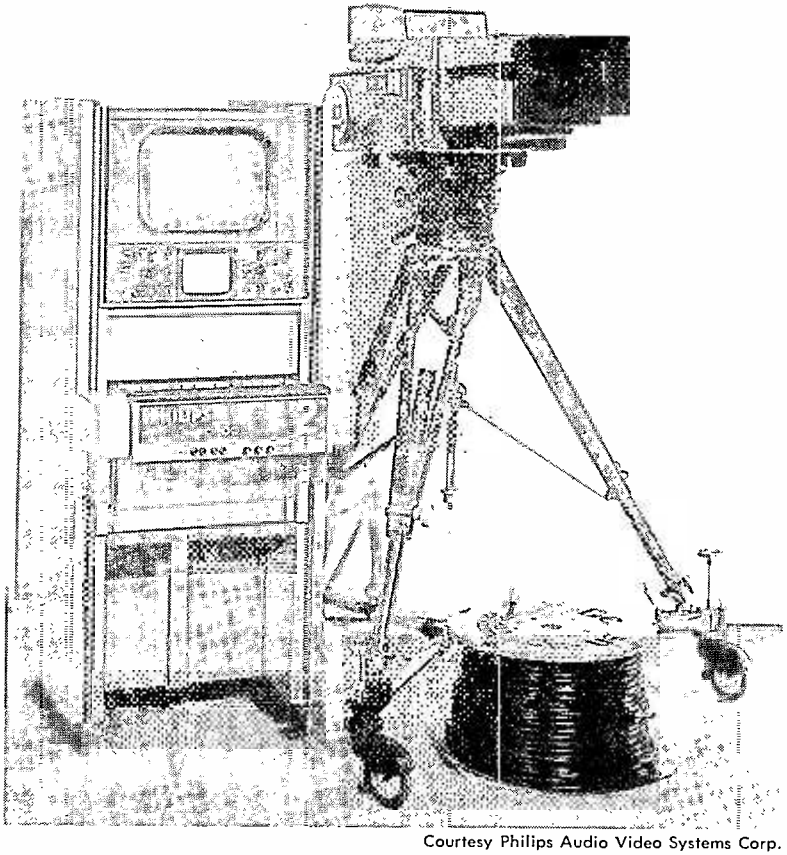
(*B*) *System block diagram.*

Courtesy Philips Audio Video Systems Corp.

Fig. 9-34. PCP-90 digitally controlled portable color camera.

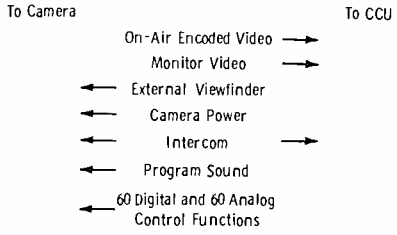
from a base station located as far as 10 miles away, depending on the transmission path.

The camera can be linked to its base station by a triaxial cable if terrain features interfere with wireless communication. However, cable losses limit the distance between camera and station to one mile, unless repeaters are used. For on-the-spot recording with a portable video recorder at the camera location, a local control box



Courtesy Philips Audio Video Systems Corp.

(A) Camera and studio control unit.



(B) Signals multiplexed on triaxial cable.

Fig. 9-35. PC-100 digitally controlled studio color camera.

plugged into the backpack allows the operator to perform all functions of the digital command system.

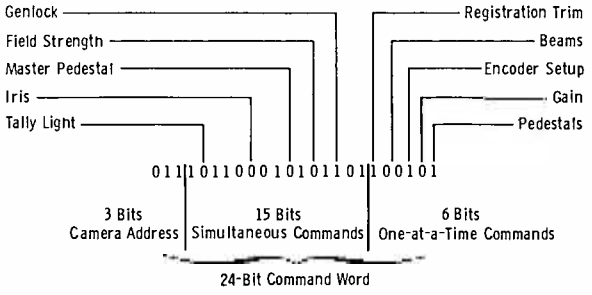
The top section of the backpack (Fig. 9-34A) contains the uhf-vhf data receiver and microwave transmitter (in the wireless ver-

sion). The main center section houses video processing, including the NTSC encoder, sync circuits, and command control circuits. At the bottom is the battery pack for wireless use, or the cable power connector.

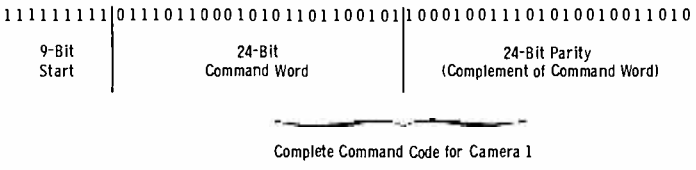
For the digital command system, frequency modulation of a 950-MHz carrier is used. An audio tone at 5.4 kHz is phase-shifted 180° to identify a single bit of information. The resultant significant sidebands occur between 2.7 kHz and 8.1 kHz on a 450-kHz command subcarrier. This permits adding a 250- to 2500-Hz interphone voice channel to the main carrier without excessive cross talk.

NOTE: With triaxial cable, different multiplexing frequencies are used, as described later in this section.

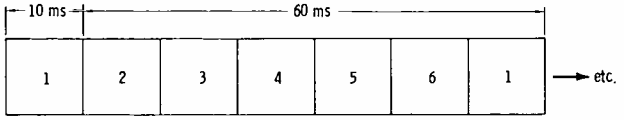
An example of a 24-bit command word is shown in Fig. 9-36A. Since up to six cameras can be controlled, the first three bits address the camera to be commanded in the immediately following word. This is followed by 15 bits of simultaneous commands, and then by six bits of one-at-a-time commands that allow any one of 64 indi-



(A) *24-bit word.*



(B) *Complete command.*



(C) *Six-camera sequence.*

Fig. 9-36. Camera command code.

vidual functions to be controlled by the base-station operator. The complete command word for each camera is preceded by a 9-bit code of all 1's (Fig. 9-36B) to alert the camera decoder that an event is to take place. Also, the 24-bit command word is followed by a 24-bit parity, which is the complement (interchanged 1's and 0's) of the preceding command word. All of this is done for the purpose of reliable noise immunity of control. Thus, unless the command word is followed by the complement, nothing happens. This virtually eliminates false commands and erroneous functions of the command system.

The complete command word may be observed to consist of 57 bits (9-bit start plus 24-bit command plus 24-bit parity). The complete 57-bit command requires 10 milliseconds to transmit (Fig. 9-36C). At the end of this time, the base-station sequencer steps to the address of the next camera. Thus, the sequence time for six cameras is 60 milliseconds, allowing each camera control an interval of 10 milliseconds 16 times during each second.

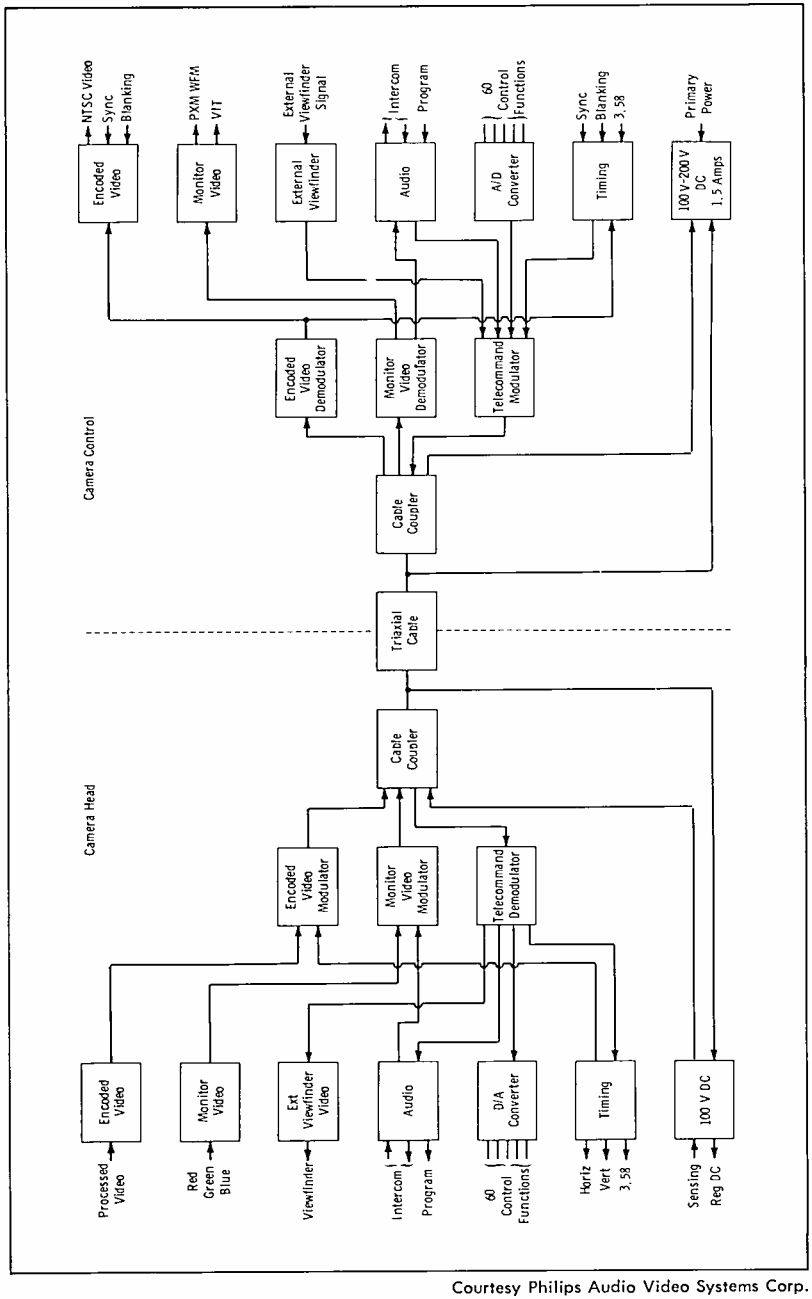
A genlock module (horizontal, vertical, and color-subcarrier lock) is used to compare local sync with sync from each remote camera. Digital commands are transmitted to bring all cameras into exact synchronization with each other and with the local sync generator. This permits fades, lap dissolves, and special effects between any combination of remote and local signals.

Philips PC-100 Camera

It is quite natural that the principles of this design be extended into the area of studio operations. Such is the case with the Philips PC-100 illustrated in Fig. 9-35. The camera and studio control unit are shown; there is also a portable control unit.

Since the video-processing circuitry is in the camera head, the effects of temperature variations on a long camera cable do not influence the camera performance. With the exception of some presets, the camera does not contain any setup controls, so that the complete lineup of the camera chain can be carried out by the camera-control-unit (CCU) operator.

Information is transferred between the camera and the CCU by a triaxial cable; this cable weighs one-tenth as much as standard color cable. Transmission is accomplished by multiplexing three channels of information through the cable (Fig. 9-37). These channels include a video channel, for sending encoded video from the camera to the control location; a monitor channel, for sending monitor signals from various points in the video-processing chain to the control location; and a telecommand channel, for transmitting all control, registration, and setup signals from the CCU to the camera. In addition to these three channels, 100-volt dc power is supplied



Courtesy Philips Audio Video Systems Corp.

Fig. 9-37. Simplified block diagram of the PC-100 system.

to the camera through the triaxial cable. The maximum cable length is one mile (more with the addition of repeaters).

The electronic viewfinder contains a 7-inch rectangular picture tube. It is tiltable, rotatable, and removable. Any combination of either RGB, minus G, or Y and external video signals can be selected for display. An electronic zoom indicator is superimposed on the top of the picture. The "on-air" tally light can be seen from all angles.

The camera control unit is separated into three major subassemblies: the monitor unit, the registration and operating panel, and the electronics unit. These subassemblies are linked by cables in the rear of the console and can be accommodated either in a standard 19-inch rack or in two 19-inch transit cases.

The electronics unit consists of a 7-inch-high rack that houses a two-level card bin. The card bin is mounted in a retractable, tiltable drawer for ease of servicing. Coaxial and multipin connectors are mounted on the rear panel. The circuitry includes the cable-drive demodulators, audio modulators, the analog-to-digital (a/d) converters, and the power supply. Final processing of the encoded video signal occurs in the electronics unit with the addition of studio sync and timing.

The registration and operating panels are mounted in a drawer assembly that can be placed in the cabinet at various positions. An overlay panel is provided to cover the registration controls after setup of the camera. To aid in camera setup, switches and associated lamps are interlocked to reduce the number of manual operations required. For example, a REGISTRATION push button is provided to switch the matrix and contours to off and to switch the encoder input signals to the waveform monitor. At the same time, it also switches the waveform monitor to the RGB sequential mode and applies minus-green video to the picture monitor.

As noted in Fig. 9-37, all normal features of monitoring and control are carried out at the CCU. The three main controls (telecommand, video, and monitor) are multiplexed on the cable in addition to the camera 100 volts dc and the audio signals. The frequency bands used to carry the multiplexed information on the triaxial cable are shown in Fig. 9-38. The basic function of each channel is described briefly in the following paragraphs.

Telecommand Channel—The control system consists of a/d converters at the CCU and d/a converters in the camera. Control functions in the CCU are normalized from 0 to 5 volts. Control information is carried in 8-bit digital form; each function is sampled four times per field, or 240 times per second. The digital train (Fig. 9-39) is added to the composite external viewfinder signal, taking the place of the subcarrier burst. (The camera viewfinder is mono-

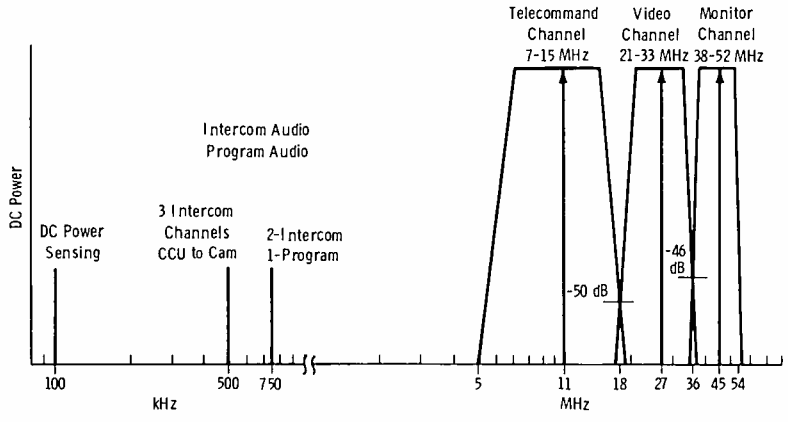


Fig. 9-38. Multiplexed channels for triaxial cable.

chrome.) This signal is then modulated on an 11-MHz subcarrier (see Fig. 9-38).

In the camera head, the d/a converter IC accepts the coded digital pulse train and converts it to the exact analog voltage generated by the potentiometer on the panel of the CCU. One IC per channel is plugged into a single module whose outputs drive all camera controls.

In the d/a converter, serial digital data are processed into a decoding register. A clock pulse synchronized to that in the CCU directs the signal to the proper d/a decoding function. During the next horizontal period, eight parallel data bits are clocked out of the register into a storage register. The output of a buffer drives a set of matched ladder switches, completing the transition back to an analog voltage. (Review Section 7-9.)

Video Channel—The video system produces a composite color signal, which, as shown in Fig. 9-38, modulates a 27-MHz carrier. A line amplifier is used to provide the required output drive capability.

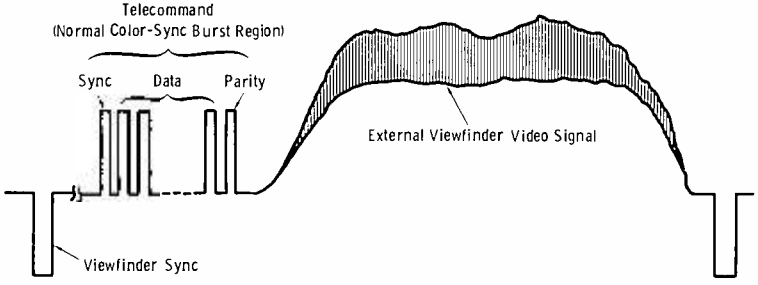


Fig. 9-39. Simplified telecommand-channel waveform.

Monitor Channel—Camera video signals are selected at various points in the chain by a video switch that receives its commands from the data control link. Vertical-interval coding is contained on lines 12 to 21 of the vertical-blanking interval to provide data on the camera performance status, such as power and scan, and to transmit the call signal. The composite monitor signal is then applied to a 45-MHz carrier for transmission to the CCU, as indicated in Fig. 9-38.

Audio—Two-way intercom and program sound are provided between camera and CCU. As indicated in Fig. 9-38, amplitude modulation of three separate carriers near 500 kHz from CCU to camera and 750 kHz from camera to CCU is employed for audio transmission.

Chroma Key—A chroma-key signal is generated in the camera head and brought back to the CCU on the encoded video signal. When the chroma-key switch is depressed, a positive-going pulse is generated whose width and timing are dependent on the color selected at the control point and the location of this color in the scene. A hue control selects the color in the scene to be keyed. A level detector in the CCU separates this pulse from the video signal so that the main camera can be switched off for the pulse duration. In this way, the keyed object is inserted into the main-camera video signal.

Power Supply—The control location provides the dc power for the camera. For safety reasons and for automatic cable-length compensation, a sensing signal is sent back to the power-supply regulator in the CCU. As shown in Fig. 9-38, the signal is sent back in the form of a 100-kHz frequency-modulated carrier. When the main power is shut down, a low-current power source remains on to power the audio intercom system.

The Ikegami Digitally Controlled Color Camera

The burgeoning field of electronic news gathering (ENG) has been made possible by the development of hand-held miniature color camera systems. The Ikegami HL-35 hand-held color camera head is illustrated in Fig. 9-40A. Either cable or wireless operation is possible with a backpack. Fig. 9-40B shows the wireless application, and Fig. 9-41 is the complete block diagram for wire or wireless operation. Either ac or dc power may be used for the backpack and camera head.

With the digital command technique, a single coaxial cable up to 1500 meters (4900 feet) long interconnects the backpack unit and the base station. The cable carries video, horizontal and vertical slip, horizontal and vertical centering control, iris control, tally on-off, and return video.

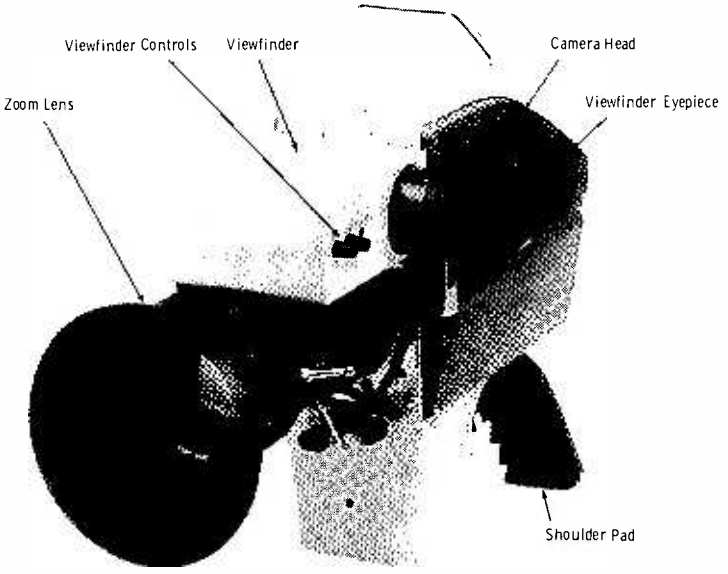
The basic system consists of the camera head and the backpack. This system is for use in the studio or in connection with a portable vtr. The backpack contains a battery pack for use in the field. From the backpack, composite color signals (NTSC or PAL) can be obtained. Provisions are made to allow the cameraman to control the start and stop of a video tape recorder remotely.

In addition to the basic system, a cable adapter is mounted on the top of the backpack, and an auxiliary unit is connected at the base station.

The cable adapter modulates video, program audio, and intercom audio. Also, the cable adapter demodulates return video, digital command signals, and intercom audio from the auxiliary unit.

For wireless operation, a carrier in the 450-MHz band is used to relay the command signal from the base station to the camera backpack. This includes color lock, horizontal and vertical slip and centering, and intercom signals. Transmission is by frequency modulation with a maximum deviation of ±40 kHz. A whip antenna is employed at both transmitter and receiver for the uhf signal.

The command signal is phase-shift keying of a 4500-Hz tone. The clock frequency is 2.4 kHz. A word is formed from 26 bits, with a frame occupying eight words. Five words in one frame are used for transmission of color-lock data. Command controls are as follows:



Courtesy Ikegami Electronics (USA) Inc.

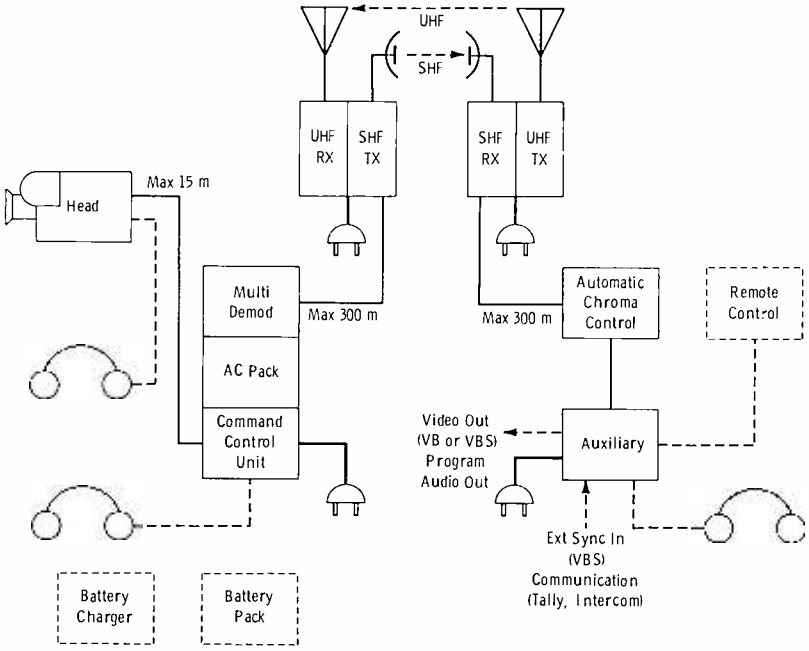
(*A*) *Camera head.*

Fig. 9-40. Model

(A)	Color Lock	Analog	8 bits
(B)	Horiz Slip A	On-Off	1 bit
(C)	Horiz Slip B	On-Off	1 bit
(D)	Vert Slip A	On-Off	1 bit
(E)	Vert Slip B	On-Off	1 bit
(F)	Iris	Analog	8 bits
(G)	Master Pedestal	Analog	6 bits
(H)	Red Horiz Cent	Analog	6 bits
(I)	Red Vert Cent	Analog	6 bits
(J)	Blue Horiz Cent	Analog	6 bits
(K)	Blue Vert Cent	Analog	6 bits
(L)	Tally	On-Off	1 bit
(M)	TX Standby	On-Off	1 bit

Note that the digital signals are converted to a varying analog signal (as, for example, to control fine centering) or to an on-off switching signal.

Transmission between the backpack (camera) location and the base station is by superhigh frequency (shf, or microwave) signals, normally in the 7-GHz range. Other microwave frequencies are avail-



RX = Receiver; TX = Transmitter; VB = Video With Blanking; VBS = Video With Blanking and Sync

(B) Diagram of wireless operation.

HL-35 color camera.

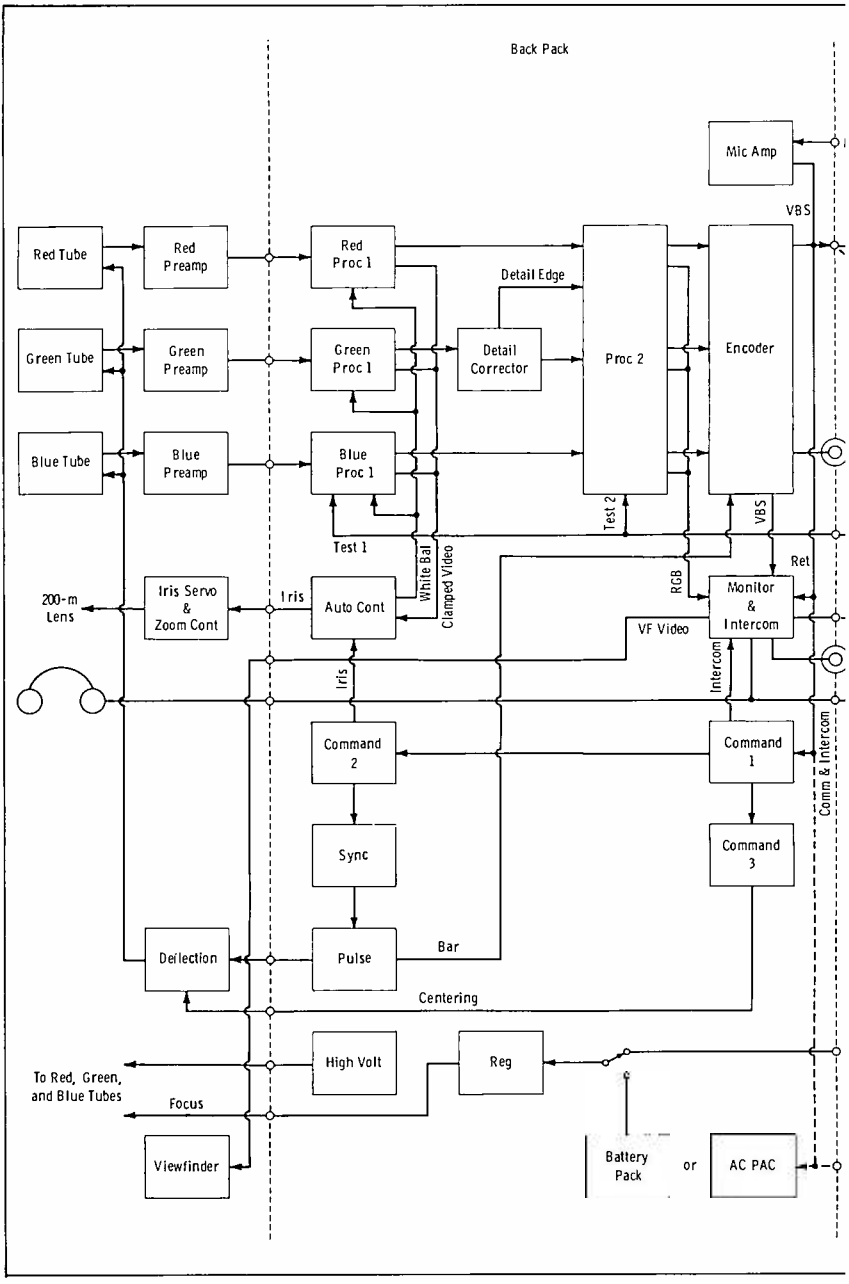


Fig. 9-41. System block

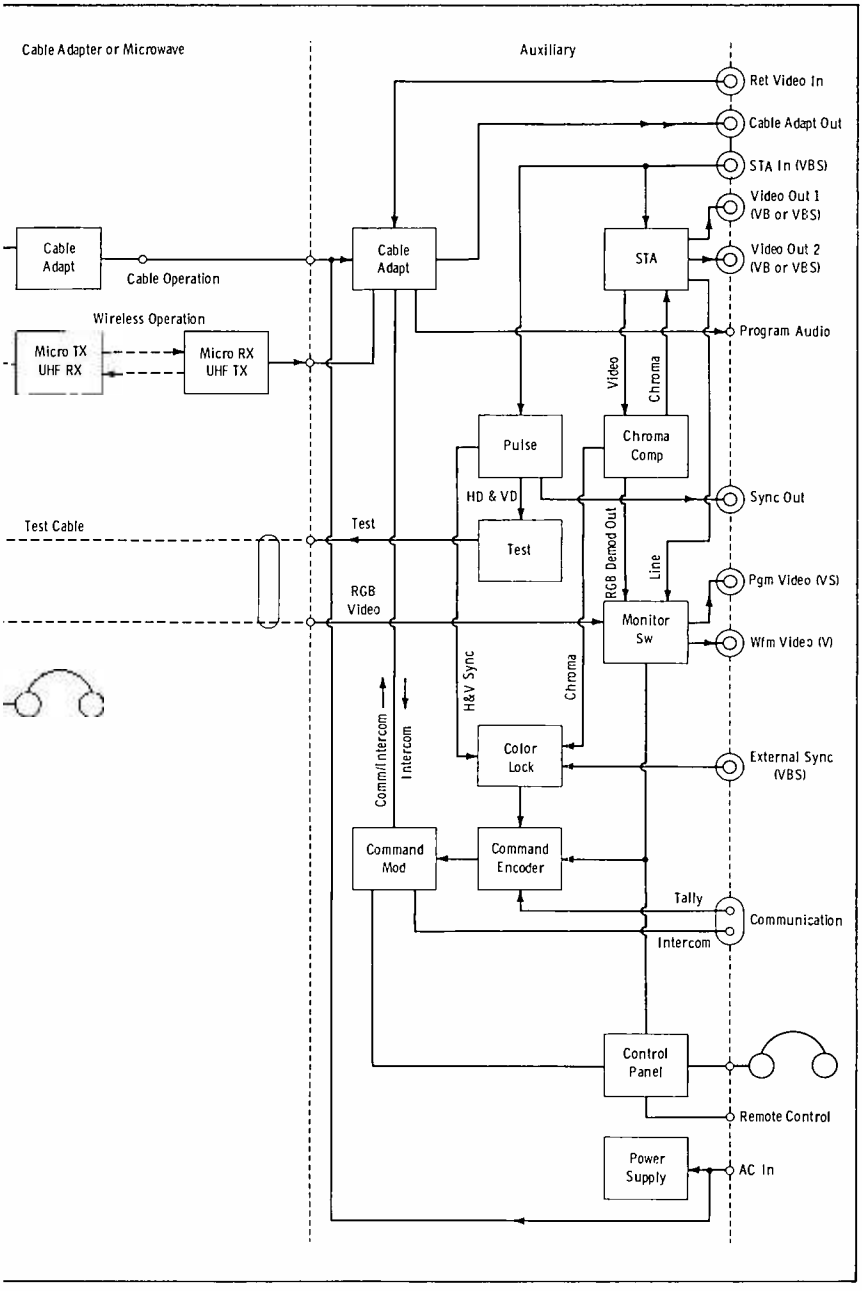


diagram of Model HL-35 camera.

able. This signal carries the video signal and program and intercom audio. The video input frequency modulates the main carrier. Program and intercom sound are carried on two subcarriers, one at 6.8 MHz and the other at 7.5 MHz. Parabolic antennas are used for the shf signals.

A functional description of the system block diagram (Fig. 9-41) follows.

Backpack—The backpack assembly consists of the processing amplifiers and the battery pack. The processing amplifier includes the detail corrector (horizontal and vertical aperture correction), the automatic controller, the command controller, the color encoder, and the sync generator.

Detail Corrector—The detail corrector is an image enhancer in which an ultrasonic delaying element is applied. The signal-to-noise ratio is improved by a comb filter.

Automatic Control—The automatic control circuit provides automatic iris control and automatic white balance. The automatic iris control circuit controls the iris servo by detecting the peak level and apl from the RGB video signal.

The automatic white balance circuit detects the white peak level of the video signal and controls the gain of the processing amplifier. This signal is "memorized" by pushing a button on the rear panel.

Optical Black—An optical black clamping circuit is provided in the processing amplifier. It clamps to the pedestal setting point after detecting the black level of a mask that is inserted at the optical assembly. Therefore, flare caused by pickup tubes or the zoom lens is completely corrected, and the black level is kept continuously stable.

Command Control—A digital signal is sent out from the base station and is converted to an analog signal or an on-off switching signal.

White-Level Clipper—A white peak clipper set to 105% white level is incorporated in the processing amplifier. It is adjustable from 80% to 200% white level. A knee circuit is incorporated for expanding the dynamic range of the video amplifier by compressing the white level. The knee point is adjustable within the range from 70% to 200% white level. Also, the gradient of compression can be set.

Auxiliary Unit—The auxiliary unit operates in the opposite mode from the cable adapter; when the auxiliary unit modulates, the cable adapter demodulates. The video signal fed from the backpack is precisely color locked by means of digital commands and is synchronized with other video programs at the base station. The auxiliary unit consists of cable-adapter, STA, chroma-compensator, color-lock, command, test-generator, and monitor-switcher units.

Cable-Adapter Unit—The cable-adapter unit is used for modulation and demodulation of the multiplex signal in the cable operation mode.

STA—The STA unit is a simple stabilizing amplifier used for clamping of the video signal and separation of the sync signal at the blanker. Also, the sync signal can be added with the STA.

Chroma Compensator—The chroma compensator demodulates the chroma signal and adjusts both gain and pedestal level. Then it remodulates the chroma signal and adds it to the line output signal of the STA.

Color Lock—The color-lock circuit compares the phase of the external sync (VBS) signal and the video signal from the backpack. Through a command, it locks the sync generator of the backpack to the external sync signal.

Command—The control signals—such as color lock, iris, tally, horizontal centering, vertical centering, and TX standby—are converted to digital signals and sent by phase-shift keying to the command unit in the backpack.

Test Generator—The test generator produces two kinds of test pulses for setting up. These signals are fed to the backpack through a test cable.

Monitor Switcher—The monitor switcher routes the output of the RGB processing amplifier to a picture monitor or a waveform monitor.

EXERCISES

Q9-1. In a digital control system, is analog information converted to digital information?

Q9-2. What is the basic difference between digital control and the type of control that employs tones?

Q9-3. How many control functions will a 5-bit binary word provide?

Q9-4. How many bits are required for two control functions, such as raise and lower?

Q9-5. How can subaudible telemetry signals (below 50 Hz) be used on a landline, when such lines typically cut off below 300 Hz?

Q9-6. How can a dc value be converted to an ac value for transmission?

Q9-7. What method of modulation is used to convert a single tone frequency to binary 1 or 0?

Q9-8. What determines the resolution of a character generator?

Q9-9. What determines the maximum frequency of the master clock in a video character generator?

Q9-10. Basically, what is "programmed electronic editing"?

CHAPTER 10

Digitized Audio and Video Techniques

The burgeoning field of digitized audio and video is rapidly becoming very important to broadcasting, recording, and general-communications technicians. This technique involves the actual conversion of analog signals (aural, visual, or other signals) into a digital form and the reconversion from digital to analog form.

The inevitable question is, "*Why* would anyone want to convert an analog signal to digital information, when it must then be reconverted to analog form?" The simple answer is that this process is needless *unless* some form of *storage* is required for proper processing of the signal for special use, or some form of *multiplexing* is required that cannot be done at all in the analog domain.

The only way an analog signal can be "stored" is by recording on tape, wire, disc, or some other medium. Such storage is always sequential; the "end" is always a given number of seconds, minutes, or hours away from the "start." Furthermore, in the process of analog storage, a certain degree of impairment of the original signal inevitably occurs.

When the analog information is converted to digital form, the digital form may be conveniently stored, and it may be read out on any desired time base. Digital pulses are not "distorted," and the possibilities for processing in digital form are virtually limitless. Within the design limitations, the digital system provides strictly "hands-off" automatic operation.

Present audio applications are:

1. Control of "acoustical size" in sound recording, as well as the "liveness" of the simulated musical environment.
2. Generation of pseudostereo and quadraphonic sources.
3. Audio echo and trick effects.
4. Acoustical synchronization of sound from multiple speaker locations in auditoriums, theaters, convention centers, stadiums, churches, outdoor installations, or other large areas.
5. Combined audio-with-video transmission system to permit network audio to be combined with network video for distribution nationwide over a single transmission facility. (Totally separate facilities are still extensively used to carry audio and video.)
6. Transmission of two wideband sound channels within the blanking intervals of a composite color television signal.

Present video applications are:

1. Automatic raster and color synchronizers and timers
2. Time-base correctors for all forms of video recorders
3. Image enhancers
4. Fully automatic standards conversion
5. Video compressors and expanders
6. Special effects providing automatic synchronized mixing of any signal(s) regardless of origin
7. All-electronic slide projectors (electronic still storage)

All of the above applications require some form of temporary storage. This is the primary reason why the digital system lends itself to applications that cannot be done as well, or at all, by strictly analog means. While systems vary considerably depending upon application and manufacturer, the basic hardware is as follows:

1. An interface unit between the analog input(s) and the digital system. This unit may or may not provide any necessary processing of the analog signal to adapt it for digital processing.
2. A low-pass filter to prevent any signal component from passing through that approaches one-half the sampling rate of the specific system.
3. A sample-and-hold circuit to convert the analog signal to a pulse-amplitude-modulated (pam) signal, and to hold each instantaneously sampled level for a sufficient time to allow generation of a corresponding binary number.
4. An analog-to-digital converter (adc) that generates the binary number corresponding to the instantaneously sampled level value. The signal is now pulse-code modulated (pcm).
5. A memory (storage) unit for read-in of the data at a clocked rate.

6. A clock generator. For audio, this is generally a tightly controlled "stand-alone" pulse generator of constant frequency and negligible jitter. For video, the clock is normally driven from the local sync generator. The clock frequency is used to read out the stored information from the memory upon demand.
7. A digital-to-analog converter (dac) that converts the binary-coded signal to corresponding analog values.
8. Another low-pass filter to eliminate all frequencies above the intended passband that are generated by switching in the digital processing.
9. An output processing unit to interface with the lines or equipment to be fed.

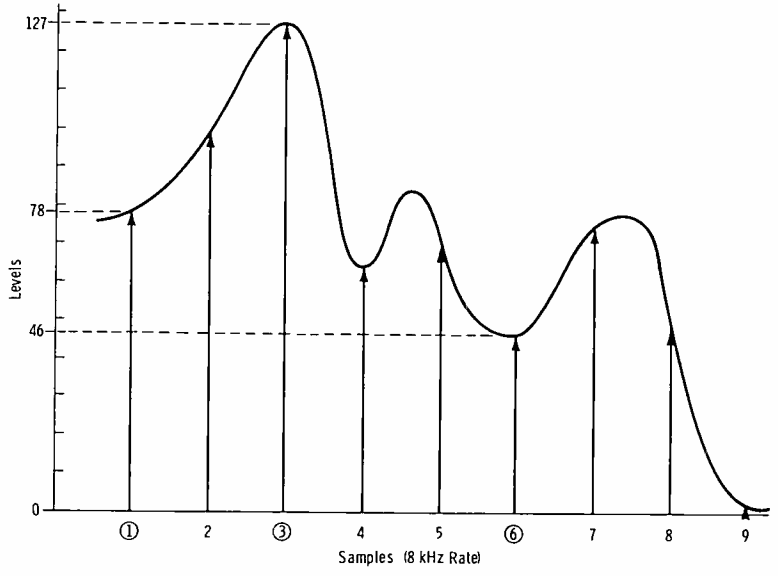
Section 1-4 ("Basic Digital System Theory"), Section 1-7 ("Digitized Audio and Video"), and Section 8-7 ("Pulse-Code Modulation") should be reviewed before you go ahead with this chapter. Due to the very recent appearance of the subject matter, an extensive bibliography is included at the end of this chapter. It will be noted that the most abundant source of references is the *Journal of the Society of Motion Picture and Television Engineers* (SMPTE) in issues published after 1973.

10-1. BASIC DIGITAL AUDIO TECHNIQUE

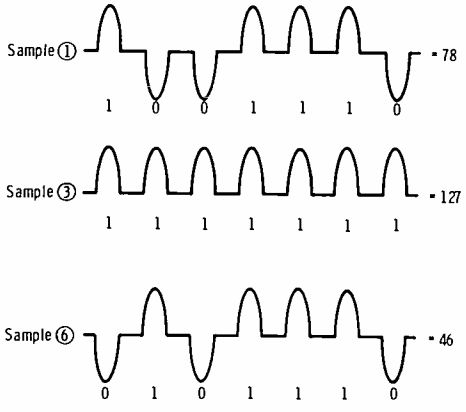
In the coding shown in Fig. 10-1, negative pulses represent binary 0 and positive pulses represent binary 1. This coding is used in some digitized audio systems. There are many ways to implement a binary code, and systems may use any of the codes already described in this book.

Fig. 10-1 shows the basic principle of a digitized voice channel with a frequency response to 3000 Hz. The sampling rate of 8 kHz is well above the Nyquist minimum. Each instantaneously sampled level is converted to a 7-bit binary word, which allows resolving $2^7 = 128$ levels, including zero. If a 10-volt peak audio level is assumed, the levels-to-voltage correlation is as shown by the table in Fig. 10-1B for the sampled levels in the example. In this example, each quantum (least significant bit) is 0.078 volt. Since the maximum error is $\frac{1}{2}Q$, the maximum quantizing error is ±39 millivolts, or 0.39% of a 10-volt signal.

If a 10-volt signal (peak amplitude) is assumed to be the maximum (full-load) input level, let us see what happens when the audio level changes to a lower value. Each quantum still has a value of 78 mV; this is fixed by the number of bits (seven) making up the binary word. Therefore, when the audio level falls to (say) 1 volt, the maximum quantizing error remains at 39 mV. This is now



(*A*) *Voice waveform.*



Level	Voltage
127	10 V
78	6.14 V
46	3.62 V
0	0 V

(*B*) *Samples.*

Fig. 10-1. Example of digitized voice channel.

3.9% of the 1-volt level. Thus, a digital system involving adc is at its best under the full-load (maximum-input) condition, which is just the inverse of the case for an analog system.

Signal-to-Distortion Ratio (SDR)

Recall from previous coverage that quantizing error and quantizing noise are generally taken to be one and the same. Similarly,

quantizing error can be related to distortion, which has a rough relationship to total harmonic distortion (thd). These relationships give rise to the *signal-to-distortion ratio (sdr).*

The sdr is dependent on the number of bits assigned to each binary word as follows:

$$\text{sdr} = 6n + 1.8 \text{ dB}$$

where,

sdr is the signal-to-quantizing distortion ratio (rms) at maximum input level (full load),

n is the number of bits in each sample word.

For the example of Fig. 10-1 (7-bit binary words):

$$\begin{aligned}\text{sdr} &= (6)(7) + 1.8 \text{ dB} \\ &= 42 + 1.8 \text{ dB} \\ &= 43.8 \text{ dB}\end{aligned}$$

This means that at the maximum input sine-wave level, the ratio of signal to rms quantizing distortion is 43.8 dB for a 7-bit binary word.

Since the value of sdr is entirely new to the average communications engineer, it is helpful to relate sdr to the more familiar total harmonic distortion (thd). Since there is no direct relationship, and since sdr and thd are not identical impairments, only a very rough relationship can be established (Fig. 10-2).

Note from Fig. 10-2 that an sdr of 43.8 dB is roughly equivalent to a thd of 1.2% at full load. This is, of course, a highly acceptable amount of distortion for voice-band reproduction. Furthermore, voice applications generally include compression amplifiers so that voice levels are held at full-load inputs, and dynamic range is not a problem.

Program signals involving music require more exacting circuit performance. Even with a compressed dynamic range at the sending end (with corresponding expansion of dynamic range at the receiving end), a dynamic range of about 40 dB is necessary. The digital system is no longer operating at full load all the time.

For example, when only 7-bit binary words are used, the 43.8-dB sdr becomes only 3.8 dB on passages 40 dB below full level. The noise level and equivalent thd become intolerable on "quiet" program transmission facilities. The actual subjective effect depends on the s/n ratio of the program circuitry or transmission path.

Because of these considerations, digitized audio normally employs 12- to 14-bit binary words, where each word describes the corresponding analog level. For a 14-bit word, the sdr is:

$$\text{sdr} = (6)(14) + 1.8 \text{ dB}$$

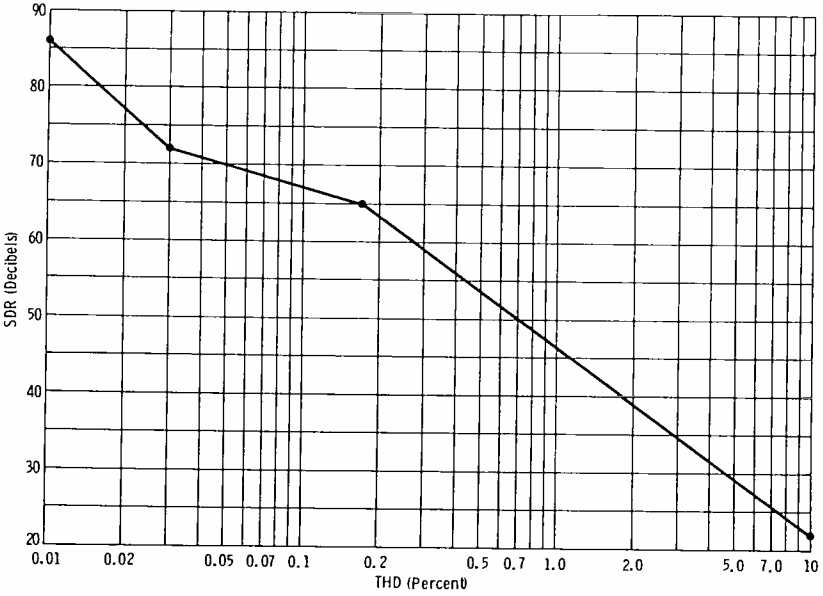


Fig. 10-2. Rough relationship between sdr and thd.

$$= 84 + 1.8 \text{ dB}$$
$$= 85.8 \text{ dB}$$

From Fig. 10-2, this is (very roughly) equivalent to 0.01% thd. Thus, the full-load sdr is at an extremely high performance level. Now consider the value of sdr when the signal is 40 dB below full level. The value is now 85.8 − 40 = 45.8 dB. This is something near 1% thd, still satisfactory for a program circuit.

Headroom

The term *headroom* is generally applied to audio transmission to indicate the maximum level under a limiting or distortion level. Thus, if a given system is to have a headroom of 10 dB, this means that the maximum audio level (peak) is to be held at −10 dB, where zero reference is the limiting value or the level for a given distortion value.

Sampling Rate

The sampling rate for a signal with a base frequency range to 15 kHz is generally at least 10% more than twice the highest frequency. The minimum sampling rate therefore becomes about 33 kHz. The transmission rate is the product of the sampling rate and the number

of bits in each binary word. Then the transmission rate for this example is:

$$\text{Transmission rate} = 33{,}000(14)$$
$$= 462 \text{ kilobits per second}$$

Required Bandwidth

Shannon's theorem (Section 1-4) permits calculation of the required bandwidth for a given channel capacity (transmission speed) and s/n ratio. Assume the transmission rate is 462,000 bits/second and the s/n ratio is 40 dB. Then:

$$\text{Bandwidth} = \frac{\text{Channel capacity}}{\log_2 \text{s/n}}$$
$$= \frac{462 \text{ kilobits}}{\log_2 10{,}000}$$
$$= \frac{462 \text{ kilobits}}{13.2}$$
$$= 35 \text{ kHz}$$

NOTE: 40 dB = 10,000 power ratio. Use graph in Appendix C to find $\log_2$ 10,000.

Thus, a bandwidth of about 35 kHz is required for a digitized audio signal with a bandwidth to 15 kHz, if 14-bit binary words are used. This shows that a wider bandwidth than the baseband audio-frequency gamut is required. The actual bandwidth required is related to the sampling frequency and the number of bits in each binary word. Thus, if only 8-bit words were used in the above example, a bandwidth of about 20 kHz would be required. However, 14-bit words provide 16,384 possible quantizing levels, allowing extremely fine resolution (accuracy) of audio reproduction. In practice, 10- to 12-bit binary words have been used for digitized audio with highly satisfactory results.

Digital Companding

A device known as a *compander* is sometimes used to reduce sample word length and hence to reduce required bandwidth. This is important when long transmission paths of wire, cable, or radio relay are involved. In the case of a radio relay, a reduction from 14 to 12 bits permits a sizeable reduction in subcarrier bandwidth.

A compander essentially compresses the data signal at the originating end and then expands the signal at the receiving terminal. As an example, the compander may truncate the three most significant bits during soft passages, since the MSBs carry no intelligence

during this time. Conversely, the three least significant bits may be eliminated during loud passages. When such action occurs, one bit in the word must be a "law" bit so that the decoder knows what has been done to the received word. Automatic circuitry then restores whatever information was deleted. Thus, truncating three bits gives an effective reduction of only two information bits, but an advantageous reduction in required bandwidth has been obtained.

Shannon's theorem gives the maximum theoretical channel capacity in terms of bandwidth and s/n ratio, but it does not fix the type of encoding to be used. The actual information rate is determined by the efficiency of a given code. (Review Section 5-7.)

The Need for Pre-Emphasis and De-Emphasis

The quantizing error inherent with pcm gives rise to a flat, wideband audio noise whose amplitude changes with program material that has a wide dynamic range. This is sometimes termed *program-modulated noise*, which may be effectively eliminated by pre-emphasis and de-emphasis of the signal prior to quantizing and the following decoding. Essentially, this reduces high-frequency noise, which is more objectionable and less easily masked by program material.

Another important benefit from pre-emphasis and de-emphasis is reduction in subjective impairment caused by sample errors. Although these are small (with a large bit-word), impulsive clicks can occur in the recovered audio signal. Terminal-point de-emphasis tends to reduce the rise time of the samples, thereby reducing subjective impairment.

10-2. DIGITAL AUDIO DELAY

One of the most common applications of digitized audio is in obtaining delay, which may be required for any number of reasons. For example, in broadcast and recording studios, digital delay units are used as signal processing for special effects, enhancement of musical sounds, instrument or vocal doubling and thickening, echo, increased ambience in acoustically flat environments, delayed feed to external reverberation units, frequency modulation and stereo or quad synthesis, etc.

Another increasingly frequent use of digital delays is in sound distribution and reinforcement systems. This permits high-quality sound to be evenly distributed without distracting echoes and loss in intelligibility which result when sound arrives from different speakers at different times. In addition, sound can be made to appear to be coming from the stage or live source, rather than from nearby speakers.

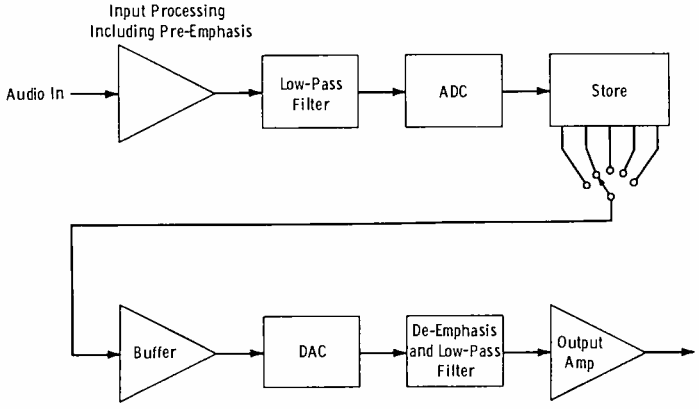
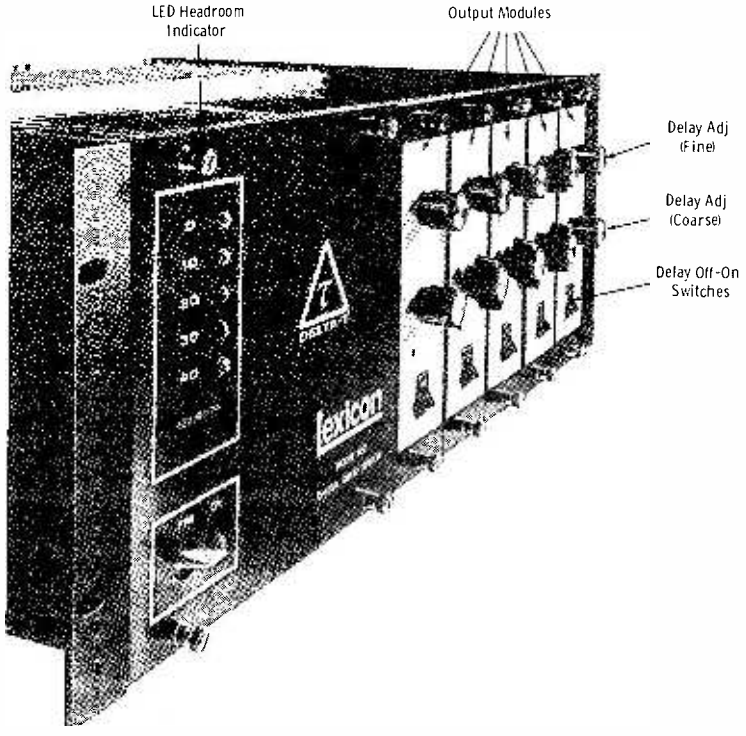


Fig. 10-3. Simplified block diagram of digital audio delay unit.

Fig. 10-3 shows a simplified block diagram of a typical audio delay unit. The input processing amplifier normally includes pre-emphasis and serves as an interface between the audio input and the adc system. The low-pass filter serves to eliminate all frequency components above the highest passband frequency, normally 15 kHz. The pre-emphasized analog signal is then converted to a digital code (pcm) in the adc, and the code pulses are fed to a series of shift registers. The outputs of the shift registers appear on a tapped switch to permit adjustment of delay. The signal leaving the switch is buffered and converted back to an analog signal in the dac. The signal is then de-emphasized on a curve complementary to the pre-emphasis, filtered to minimize transients, and fed to an output power amplifier.

The Lexicon Model 102 digital delay system is shown in Fig. 10-4. Each main frame may contain one to eight delay modules and one to five output modules. The system accepts a single high-quality audio signal at its input and delivers the same signal, but time delayed, at each of up to five outputs. Delay time for each output is individually adjustable (coarse and fine) by front-panel selector switches in up to 64 increments from zero to the maximum delay (determined by the number of delay modules installed). The amount of delay and number of outputs may be increased by cascading additional slave main frames via a digital bus.

There are three versions of the Model 102. The "A" version, generally used for sound reinforcement, has a frequency response to 12 kHz, with a delay capacity of 40–320 milliseconds. The "B" and "C" versions, normally used in broadcast and recording studios, have a frequency response to 15 kHz (±2 dB measured 14 dB below limiting). The "B" version has a delay capacity of 40–320 ms, while the "C" version has finer increments of 16–128 ms.



Courtesy Lexicon, Inc.

Fig. 10-4. Digital delay system.

A realizable dynamic range of at least 90 dB is typical. A five-position LED headroom indicator calibrated in 10-dB increments below limiting is provided.

A technical description[1] of the theory of operation of this unit will be given to yield an insight into audio delay technology.

The LED Headroom Indicator and Pre-Emphasis

In all audio systems, whether for tape recording, disc mastering, broadcasting, or mixing, some sort of level indicator is used to guide the operator in correctly setting the signal level. Correct signal level is largely defined by the need to avoid the distortion that results from overloading the medium on large signal peaks. Since most audio systems use pre-emphasis and de-emphasis to reduce the noise of the storage or transmission medium, overload is frequency dependent and related to the program spectral characteristics. Conven-

1. Courtesy Lexicon, Inc., Waltham, Mass.

tional level indicators suffer from several limitations. Generally, they monitor an analog electrical signal before storage or transmission in the medium and do not accurately reflect the manner in which the medium overloads. For example, the ballistics of a VU meter do not permit short-duration transients to be observed. Most indicators are connected before pre-emphasis and cannot reveal the possibility of overload on signals with above-average high-frequency content. Finally, media such as disc and tape have complicated overload mechanisms that are difficult to characterize in the domain of the driving signal.

The Delta-T 102 headroom indicator has several important characteristics. The incoming analog signal is pre-emphasized, filtered, and sampled in time and amplitude. The result is a stream of digital words that carry the amplitude information. These words are analyzed and decoded by logic circuits into the headroom-indicator display to give, directly in the digital domain, a precise, virtually instantaneous knowledge of the peak signal level relative to overload. The headroom indicator can thus be used with any program signal and will not mislead the user: If the "0" LED never lights, the signal has not caused overload, no matter what its spectral content or transient characteristics may be.

Since this indicator is located after pre-emphasis, the pre-emphasis need not cause the high-frequency overload problems typical of disc or tape recording or fm broadcasting. Due to pre-emphasis, music unusually rich in high-frequency content above 3 to 4 kHz may necessitate a lower operating level than conventional signals, but the 90-dB dynamic range ensures that this does not degrade the signal quality. In fact, the pre-emphasis provides an important reduction of the quantization noise and distortion inherent in a digital system. The input pre-emphasis is a modest high-frequency shelving characteristic. Fig. 10-5 shows the combined effect of input pre-emphasis and low-pass filtering in the 102-B. A mirror-image de-

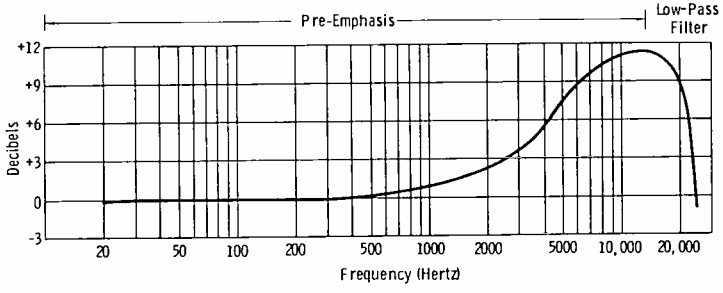


Fig. 10-5. Combined characteristic of pre-emphasis and low-pass filter.

emphasis network at the output restores the overall frequency response.

The headroom indicator has five LEDs (light-emitting diodes) arranged in a thermometer-type display. The top LED, "0," is a limit or overload indicator, as previously discussed. The other four LEDs indicate peak signal level in increments of 10 dB below the zero level. (Acutally, the LEDs correspond to levels at −12, −22, −32, and −42 dB relative to zero, but the values are rounded off to the nearest 10-dB increments for the sake of simplicity.) Each LED is driven by circuitry that keeps it on for about 100 ms.

NOTE: In the following description, terminology for the abbreviations used in the text is listed in the illustrations referred to.

General Theory of Operation

The Model 102 consists of a main frame with integral power supply and mother board, and a "library" of plug-in printed-circuit modules, or cards. In the discussion that follows, the sampling frequency, delay increment, shift-register length, etc., are those of a Model 102-B. The A and C versions differ slightly in these parameters, but circuit operation is the same.

The audio signal to be delayed is connected to an input connector and routed to the input card. There, after some analog signal processing, it is converted to a stream of 42,600 (B version) 12-bit digital words per second.

The control card performs digital operations necessary to encode the analog input into the 12-bit floating-point word. It also operates the LED headroom display and converts the parallel digital word format to a serial form suitable for passage through the delay modules.

The timing card, as its name implies, generates the basic clock signal and all the other lower clock frequencies necessary to perform error-free sampling, conversion, shifting through the delay modules, etc.,

The serial data from the control card are fed into a "pipeline" of shift registers on the delay modules and shifted one bit position at a time through thousands of storage locations in each card to achieve usable time delays. Within a delay module, the pipeline has eight taps, whose outputs are time-division multiplexed onto a single line and fed to the coarse delay selector on each output module. When selected and demultiplexed, these signals permit coarse delay selections in 40-ms increments and fine delay selection in 5-ms steps to provide 5-ms delay resolution over the entire delay time provided by the delay modules installed.

After delay-tap selection, the output module reassembles the serial data bit stream into parallel bit words which are shifted into

a digital-to-analog converter (dac). The output of the dac is gain-conditioned by the floating-point range bits, subjected to some analog signal processing, amplified, and passed to the appropriate output connector on the main frame.

The following card-by-card discussion explains system operation in more detail.

Timing Card

A crystal-controlled oscillator (Fig. 10-6) generates the basic clock rate (about 3.5 to 4.5 MHz, depending on whether the unit is an A, B, or C version). This is divided by 2, 4, and 8 in the first counter to obtain square-wave signals TS0, TS1, and TS2, respectively. The TS0, TS1 and TS2 outputs are sent to the delay and output modules, where they sequence the multiplexers and demultiplexers that manipulate the eight fine-delay taps. Pulse signals BC (bit clock) and $\overline{\text{PBC}}$ are derived from the divide-by-8 counter and are the timing signals for the serial bit stream of digitized data. The SBCG (sub bit clock gated) and $\overline{\text{PSBCG}}$ outputs are derived from the clock and occur at the clock rate, except that every 13th group of eight pulses is gated and does not appear. This signal is used in the output module to prevent shifting serial data into the serial-to-parallel conversion register during the 13th (unused) bit slot.

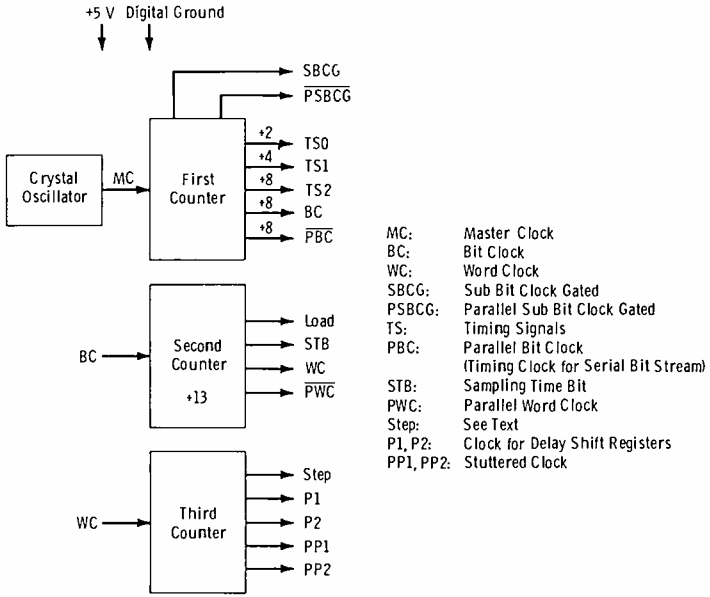


Fig. 10-6. Diagram of timing card.

Further division of BC in a second, divide-by-13 counter develops Load, WC, STB, and $\overline{\text{PWC}}$ at the word, or sampling, rate (34.08 kHz in A versions, 42.6 kHz in B and C versions). These signals time operations at the word rate, such as analog-to-digital conversion, parallel-to-serial and serial-to-parallel conversion, and digital-to-analog conversion.

More complicated counter operations occur in the third counter to derive clock pulses P1 and P2 for the delay shift registers. Although each word frame of data (at the WC rate) contains 13 time slots at the bit rate (BC), the clock-pulse stream must consist of 5-ms frames containing exactly 2561 pulses in order to tailor the 5-ms shift-register length to exactly 5 ms. This requires the clock to "stutter," sending out the first five word frames with 13 clock pulses, and the remaining 208 with 12 clock pulses. This function is accomplished by the third counter and associated circuitry. In addition, a pulse called Step is derived from the third counter at a 200-Hz repetitive rate and is used on the control card to operate the Cycle, Test mode.

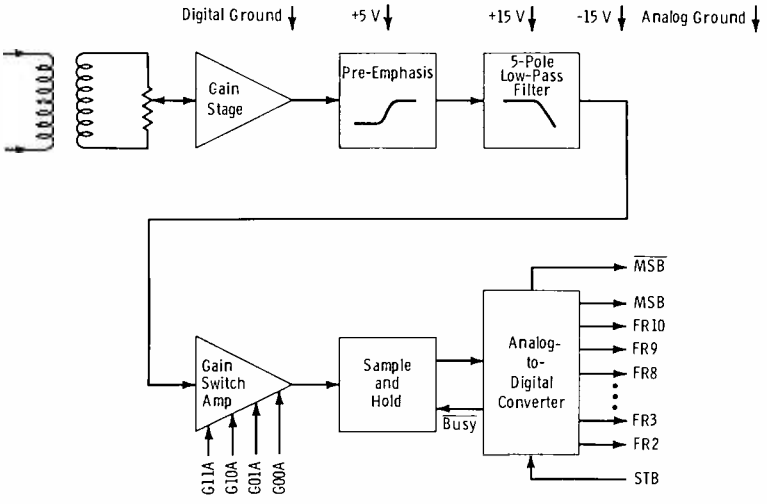
The timing card also has circuitry dedicated to use with the EX-102 extension chassis. Complements of most of the signals mentioned above are developed and appear at the mother-board pins for routing to an EX-102. Other logic steering circuitry ensures that the correct serial data signals are routed to the first delay module in this main-frame and on to the next EX-102. The serial data signal (SDZ) from the main frame is passed through as ZDSD to each output module in the main frame, as $\overline{\text{PZDZD}}$ to the next EX-102 to provide the zero-delay serial data signal for its output modules, and as SDI to the first delay module in this main frame. The output of the last delay module in the main frame appears at SDO and is passed on as $\overline{\text{SDO}}$ to the next EX-102. Thus, the next unit gets $\overline{\text{PZDZD}}$ (zero delay data) and $\overline{\text{SDO}}$ (delayed data from the last tap of the preceding main frame).

The exact clock frequency, division ratios, and stutter clock characteristics are different for the A, B, and C versions of the 102 and are programmed by jumpers and choice of crystal at the factory.

Input Card

The input card (Fig. 10-7) uses a balanced and floating transformer input to a variable gain control that adjusts the input sensitivity from +4 to +22 dBm. The transformer secondary drives an op-amp gain stage, which in turn feeds an op-amp pre-emphasis stage that provides a shelving high-frequency boost, up 3 dB at 2.1 kHz, and shelving to +12 dB at 12 kHz.

Next, the signal passes through a five-pole Butterworth low-pass filter realized with two op amps. This filter has a cutoff frequency



MSB: Most Significant Bit
FR2, FR3, etc.: Frames
G00A Through G11A: Gain Sets From Logic
STB: Sampling Time Bit

Fig. 10-7. Diagram of input card.

of 15 kHz (A version) or 18 kHz (B version) to prevent spurious input-signal frequencies higher than half the sampling rate from forming low-frequency alias tones in the sampling process.

The low-pass filter drives a fast gain-switching amplifier (gsa), whose gain is set by logic signals G00A through G11A from the control card. The gsa drives a sample-and-hold circuit (S&H) that, upon command ($\overline{\text{Busy}}$) from the analog-to-digital converter, freezes its output for a time period while the adc performs its conversion. When the 10-bit conversion is completed, the S&H quickly acquires and tracks its input signal until the next hold command. A sample-and-hold circuit is necessary here to ensure that the input waveform is sampled at precise intervals and to provide an unchanging input signal for the adc during the conversion period. Jitter in the sampling time or changing sampled-signal values during the conversion time would both be sources of error in the reconverted audio output signal if the S&H circuit were not used.

Finally, as already noted, the adc, under control of the STB signal at the sampling or word clock (WC) rate, converts each sample to a 10-bit digital word whose first bit, $\overline{\text{MSB}}$, gives the polarity or sign, and whose next nine bits, FR10 through FR2, give the quantized amplitude. The output of the card is a stream of 10-bit parallel words at the sampling rate that, when combined with two range bits on

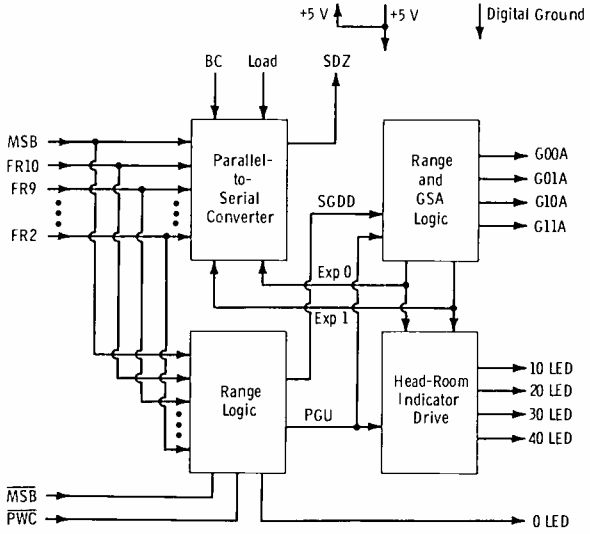
the control card, digitally encode a wideband audio input signal that has a 90-dB dynamic range.

There are five trim potentiometers on the input card. Three set the precise gain steps for the gsa. Another zeros out the offset in the gsa, and the fifth sets the adc offset so that zero input corresponds to the word 0 . . . 010. Field adjustment of these potentiometers is unnecessary and is not recommended.

Control Card

The input signal for the control card (Fig. 10-8) is the stream of 10-bit parallel words from the input card. The output signals developed are a 12-bit serial word stream, gain-control logic signals for the gsa on the input card, and the drive signals for the LED display.

The Delta-T 102 achieves its 90-dB dynamic range by using a floating-point representation of the signal amplitude, where the first two bits of the 12-bit word carry range, or gain, information and



BC: Bit Clock
MSB: Most Significant Bit
SDZ: Serial Data Signal
G00A to G11A: Gain Sets From Logic
FR: Frame
EXP: Gain Exponent Bits
SGDD: Switch Gain Down Command
PGU: Pulse Gain Up Command
PWC: Parallel Word Clock Rate
LED: Light Emitting Diode in Head Room Indicator

Fig. 10-8. Diagram of control card.

the next 10 bits carry the instantaneous amplitude. The input-card gsa and control-card logic act together as a closed-loop control system, adjusting the gsa gain in 10-dB steps to keep the average encoded signal level between one-quarter and three-quarters of the digital full-scale code. The 10-bit parallel words are examined by logic circuits to determine whether the encoded signal sample is below ¼ full scale, between ¼ and ¾ full scale, greater than ¾ full scale, or at full scale (in which case the "0," or limit, LED is lit). If the signal remains below ¼ full scale for longer than 100 ms, the command PGU is given and results in increasing the input-card gsa gain by 10 dB (unless it is already in range 4, maximum gain). The encoded signal is now 10 dB higher in level than before and, if its peak value is between ¼ and ¾ full scale, no further gain change will occur. If it is still less than ¼ full scale, and continues so for 100 ms, another PGU pulse will step the gsa gain up to the next range. There are four gain ranges: Range 4 is the maximum and is used for signals whose peak level is below about −32 dB; range 3 is for signals between −32 and −22 dB; range 2 is for signals between −22 and −12 dB; and range 1 is for signals between −12 dB and the limit.

Conversely, if the encoded peak signal level exceeds ¾ full scale, the pulse SGDD, switch gain down, is generated, and the gsa gain is immediately reduced by 10 dB. If the encoded signal level is still greater than ¾ full scale, SGDD again immediately reduces the gsa gain until the peak signal level is between ¼ and ¾ of the full-scale level, or until the gsa is in range 1 and no further gain reduction is possible.

Working constantly in this fashion, dynamically adjusting itself to the signal, the control logic seeks to utilize most of the 10-bit resolution of its adc over a 90-dB dynamic range of input signals. Sudden signal increases bring about an immediate change to a lower gain, preventing clipped initial transients. Sudden decreases in signal must be sustained for 100 ms to bring about a change to a higher gain, in order to prevent rapidly changing or low-frequency signals from modulating the gain rapidly on a cycle-by-cycle basis of the input waveform.

The control card has on its front edge three switches that are used for test and calibration and that sometimes can be helpful in diagnostic troubleshooting. The top switch selects the Normal or Test mode: In Normal, the other two switches have no effect and the 102 is in an operational mode; in Test, the gain range can be controlled independently of the input signal by the user. In the Cycle mode, the gain cycles through the four ranges in sequence, changing range every 5 ms. This is useful for setting the offset adjustment on the input card and output modules. In the Manual

mode, the small push-button switch permits controlled incrementing into any range.

The control-card gain logic also provides the gain exponent bits, Exp0 and Exp1, which are appended to the 10-bit "mantissa" word in a parallel-to-serial shift-register converter. Thus, each 10-bit word carries with it a 2-bit code indicating what gain range was used when it was formed in the adc. When decoded, these two bits are used on the output card to program a switchable attenuator that precisely complements the input-card amplifier gain setting and restores the encoded signal to its 90-dB dynamic range. This floating-point technique, with four ranges set 10-dB apart, extends the roughly 60-dB dynamic range of 10-bit encoding to 90 dB, and it provides full 10-bit resolution and corresponding 60-dB signal-to-noise ratio even at signal levels 30 dB below limiting. In contrast, a straight linear encoding of a 90-dB input dynamic range would require extremely costly 15-bit adc's and dac's, as well as additional memory for the extra three bits required. The dynamic characteristics of the gain logic, coupled with the use of pre-emphasis and de-emphasis, permit the input signal to mask the noise changes that occur with gain changes.

The headroom indicator display of five LEDs is mounted directly on the control card. As already mentioned, the "0" LED is lit whenever the 10-bit adc output reaches full scale in the first range, corresponding to limiting. The "10," "20," "30," and "40" LEDs are operated from the exponent bits Exp0 and Exp1 to indicate range 1, 2 3. or 4, respectively. The "40" LED is extinguished whenever the PGU pulse is sustained, indicating operation in range 4 with signal levels below −42 dB, calling for more gain than is available.

Connections are provided via the control-card connector to permit the use of a remote headroom indicator. This involves five LEDs connected between +5 volts and the low side of the five LEDs on the control card.

Delay Module

The serial stream of 12-bit words, SDZ, from the control card appears at the input, SI, of the first delay module (Fig. 10-9). This signal enters the first of eight segments of shift registers. At each clock pulse (derived from P1 or P2, signals from the timing card), the serial data shift one location toward the output of the shift register. By shifting data through 2561 shift-register positions, the signal is delayed 5 ms. At the end of each 5-ms segment, the signal is fed to an input of an 8-to-1 time-division multiplexer, and to the beginning of the next 5-ms segment. After the eighth segment, the signal appears as SO, serial data out. This output serves as the SI input signal for the next delay module in the main frame. The multi-

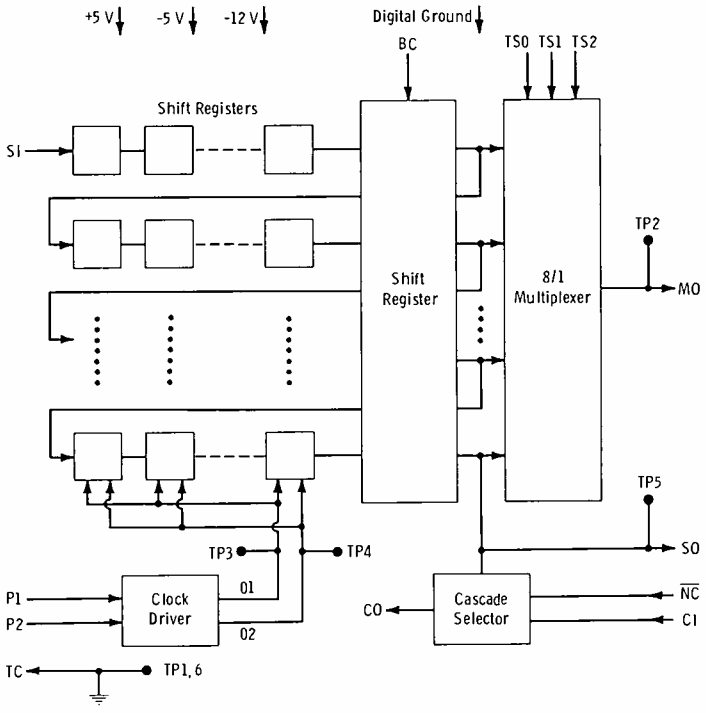


Fig. 10-9. Diagram of delay module.

plexer, with the aid of timing signals TS0, TS1, and TS2, sequences data from each of the eight taps into sequential 1-bit time slots, so that its output, MO, is actually the data for eight different time delays of the same signal. A clock-driver circuit on the card converts clock-pulse signals P1 and P2 into pulses of suitable amplitude and duration for clocking the MOS shift registers.

Additional circuitry on the delay modules is used to ensure that an extension chassis, if used, is fed data from the last (i.e., greatest) delay tap in the unit. This function is performed by a TTL selector circuit on each delay module that passes the output of the last delay tap back, in leapfrog fashion, through each delay module to the timing card. When a delay module is plugged in, its TC terminal grounds the $\overline{\text{NC}}$ terminal of the module to its left. With $\overline{\text{NC}}$ thus tied low, the module will select the signal CI and pass it on as CO to the next module on its left. With $\overline{\text{NC}}$ ungrounded, as is the case

with the last, or rightmost delay module, it selects its own output, SO, and sends that left to CI of the next module. Thus, the output of the rightmost module is sent back through the selectors on the other delay modules to the timing card, from where it will be sent to an EX-102 extension chassis.

Output Module

The output module (Fig. 10-10) performs the functions of delay-tap selection, serial-to-parallel data conversion, digital-to-analog conversion, gain conditioning, and analog signal processing.

The coarse delay control is a binary encoded switch that selects, via an eight-input/one-output IC, the output of one of the eight delay modules, MO1 through MO8. The fine delay control, in conjunction with timing signals TS0, TS1, and TS2, delivers a pulse in one of eight possible timing positions within the time of one bit-clock period, BC. This signal controls three shift registers that are converting the serial data to parallel form, selecting one of the eight delay taps of the delay module selected by the coarse control and, in effect, demultiplexing the MO data. A rocker switch selects between data at the tap selected by the coarse and fine controls and zero-delay data, ZDZD. The result is that the serial data selected are shifted into the 12-bit shift register, controlled by timing signal SBCG, and selected by the demultiplexing function of the fine control. Twelve shifts occur per word, ensuring that any spurious data in the 13th slot are thrown away.

Finally, when the 12-bit word is assembled in the shift register, it is transferred in parallel form through a 12-bit buffer. The two gain-range bits, Exp0 and Exp1, are decoded into four signals that control a MOSFET-switched attenuator which scales the dac output, while the 10 signal-amplitude bits, MSB and FR9 through FR2, are applied to the dac input lines. The dac output, after gain conditioning from the attenuator, is the restored input signal, suitably delayed. This signal passes through a buffer stage and then a five-pole, active Butterworth low-pass filter, with cutoff frequency the same as the input low-pass filter. This filter and the two-pole filter that follows reduce sampling-rate high-frequency components that come along with the desired signal from the dac. The second active filter (two-pole) provides a small amount of peaking near the upper limit of the bandwidth, to correct for high-frequency losses inherent in sampling the input waveform (aperture loss).

Another active filter provides high-frequency de-emphasis, the inverse of the input pre-emphasis, restoring the overall frequency response to its proper flatness. Finally, the signal is amplified by a small power amplifier, whose output is transformer-coupled to provide a balanced and floating output capable of delivering a +22-dBm

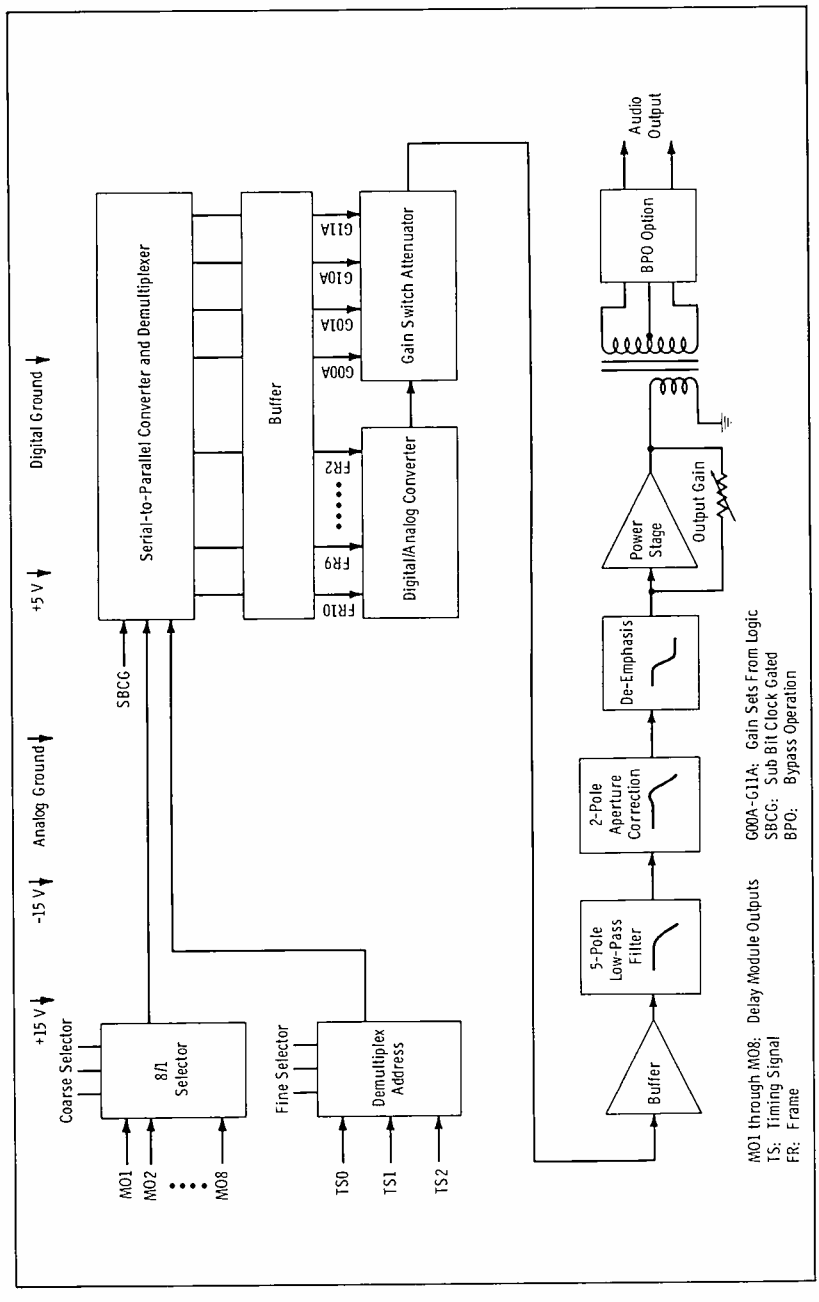


Fig. 10-10. Diagram of output module.

signal to an external 600-ohm input. A potentiometer in the feedback loop of this amplifier allows adjustment of the output level.

This module may be fitted with a dpdt relay as part of the BPO option. This relay connects the Delta-T input signal to the Delta-T output connector when power is removed. With this option and power on to the unit, and with proper operation of the +15-volt power supply, the relay closes and connects the output-module output to the Delta-T output.

This module may also be fitted with additional digital logic to permit the use of an external option, the remote delay selector. With this option, the selection of coarse and fine delay step or of zero delay can be made at a remote location.

The output module has four trim potentiometers. Three set the gsa attenuator steps precisely 10 dB apart, and the fourth is used to set the dc offset level from the word 0 . . . 010 to zero volts out of the dac. Field adjustment is not recommended, nor should it be necessary.

10-3. DIGITIZED AUDIO WITH TV TRANSMISSION

Network distribution of picture and sound must employ separate facilities for the two signals. Also, in-plant facilities of a tv station must employ separate paths for audio and video. Digital techniques provide a method of combining audio and video to eliminate a large amount of audio distribution equipment.

There are two methods currently being developed to achieve simultaneous transmission of audio and video over a common path. One method diplexes up to four 15-kHz audio channels into a single active video channel.[2] Another method transmits two 15-kHz sound channels during the horizontal-blanking interval of the television waveform.[3] The latter would obviously allow stereophonic sound for television if and when approved.

The DATE System

DATE is an acronym for *D*igital *A*udio for *TE*levision. As can be noted from the simplified block diagram of Fig. 10-11, the system converts program audio into binary data and then back into analog audio. A 5.5-MHz digital subcarrier is used to carry the binary audio data.

This system provides up to four discrete audio channels. The sampling frequency of 34.42 kHz is conveniently obtained by divid-

2. R. Evans Wetmore, "DATE, A Digital Audio System for Television," *Journal of the SMPTE*, March 1974.

3. Manfred Maegele, "Digital Transmission of Two Television Sound Channels in Horizontal Blanking," *Journal of the SMPTE*, February 1975.

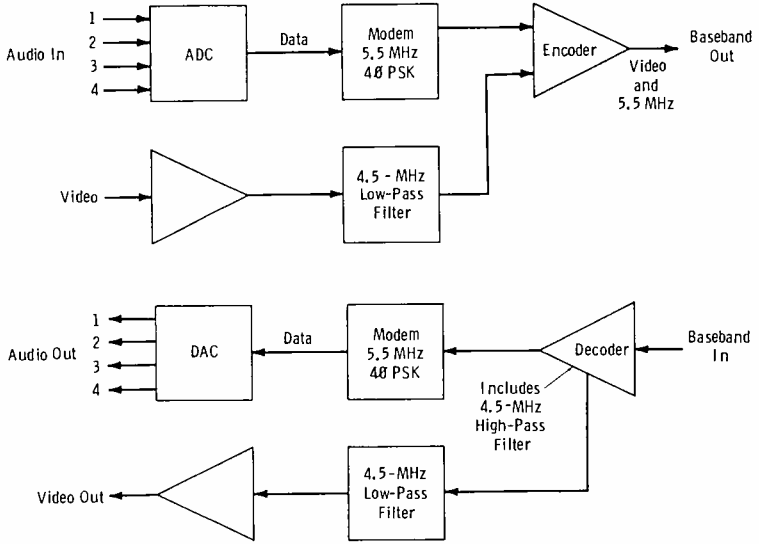


Fig. 10-11. Simplified block diagram of the DATE system.

ing the color-subcarrier frequency by 104. Each sample is transmitted as a 13-bit word; thus, the bit rate is 34.42 kHz times 13, or 447 kilobits/channel. The overall bit rate for four channels is 447 kHz times 4, or 1.79 megabits.

The 5.5-MHz subcarrier has four-phase (one phase for each channel) psk modulation. The encoder receives this signal along with a filtered video signal so that the baseband output is a 5.5-MHz subcarrier along with the video signal limited to 4.5 MHz. At the decoder, the audio and video paths are again separated by the appropriate filters, and the four separate audio channels are available along with the video channel.

Digitized Sound in Horizontal Blanking

Placing sound information within the blanking intervals of television signals has been done satisfactorily on an experimental basis. Two wideband sound channels have been successfully transmitted in the horizontal-blanking interval by digital means.[4]

There are 21 lines under vertical blanking for each field; each line has a duration of 63.5 μs. The first nine lines are occupied by equalizing and vertical-sync pulses, and the last three lines are normally occupied by national or international test signals. This leaves $21 - 12 = 9$ lines per field that would be available for some

4. Maegele, *loc. cit.*

form of audio information. This corresponds to a relative duration of 3% referred to total picture duration.

By modification of the horizontal-blanking interval, a maximum of 7.5 μs can be used for sound transmission, which is close to 12% of the line period of 63.5μs. By using the horizontal-blanking period (modified) instead of the vertical-blanking period, several advantages are obtained. First, longer periods per picture (12% compared with 3%) are made available. Second, a much shorter storage time of the sound sampling values (63.5 μs compared to 16.67 ms) is possible. Third, the required storage capacity is reduced by almost 300 times.

Fig. 10-12 illustrates the rather drastic modification required in the standard horizontal-blanking interval for this application. The front porch is reduced to about 0.6 μs from the standard 1.6 μs. Horizontal sync is reduced to 1.5 μs from the standard 4.8 μs. The color burst is shortened by two cycles and moved toward the sync pulse to shorten the standard "breezeway." It is claimed that all of these modifications do not affect receiver performance.

The above modifications allow 4.7 μs per line of digital data for two sound channels. There are 241 active lines per field, so the total sound interval is 241 times 4.7 μs, which is equal to 1132 μs per field, or 2264 μs per frame. A start pulse of 0.23 μs is used to control the digital units in the receiver. To avoid field-frequency interruption of the audio information, three additional pulses are inserted in the vertical-blanking interval. To crowd as much data as possible into the shortest practical time, either ternary coding or a binary NRZ code is used.

10-4. BASIC DIGITIZED VIDEO TECHNIQUE

The basics of digitized video have been covered in previous sections cited near the start of this chapter. It is important that the reader review these sections until the material in them is thoroughly understood.

There is a basic difference between digitized audio and digitized video that should be understood before further study. It may be recalled that the number of bits forming each sampled level is a minimum of about 10 for wideband audio at full dynamic range. For video, a synchronous type of signal occurs, since all picture information lies at even multiples of the line-scanning frequency. Color information is "interleaved" at odd multiples of one-half the line-scanning frequency. A countdown from the color subcarrier generator is used to derive all monochrome and color scanning intervals, so that a tight synchronous relationship is maintained for the entire system.

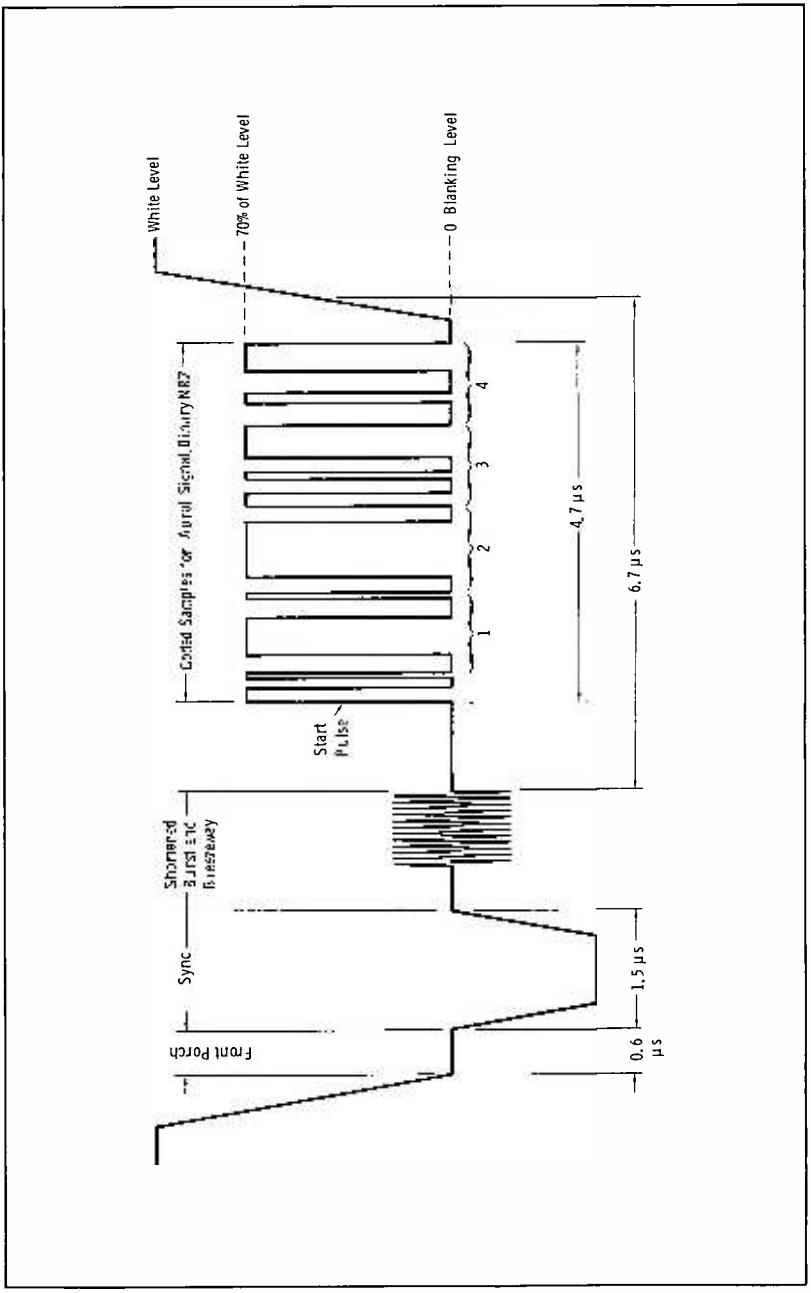


Fig. 10-12. Binary NRZ pulses integrated into horizontal blanking.

Due to the nature of the video information, and the fact that video sources have a relatively high noise factor, a smaller binary word can be used for each sampled level. In practice, an 8-bit word for each level (describing one level out of 256) is adequate for the highest quality pictures now available or anticipated for the future. Of course, due to the higher frequency content of video (to 4.5 MHz), the sampling frequency must approach 10 MHz as contrasted with about 35 kHz for digitized audio. The sampling rate is always a multiple (either 3 or 4) of the color-subcarrier frequency so that a synchronous data process is maintained.

Table 10-1 shows the average effect on the picture-monitor presentation correlated with the number of binary bits forming a word and the resulting number of sampled levels. This table indicates that an 8-bit word is entirely adequate for digitized video, since the average video source (camera, vtr, etc.) currently has a ratio of peak-to-peak signal to rms noise of about 48 dB (at best). Note that each bit reduction reduces the s/n ratio by 6 dB.

Table 10-1. Effect of Sampling Levels

Bits	No. of Levels	Comments	Unweighted Ratio of P-P Signal to RMS Noise (dB)
1	2	Can get intelligible information through, but far from broadcast quality.	13
2	4	Significant improvement over (1) but very bad contouring. Far from broadcast quality.	19
3	8	Significant improvement over (2). Still very bad contouring. Not broadcast quality.	25
4	16	Some improvement over (3). Some contouring. Not quite equivalent to present broadcast quality.	31
5	32	Small improvement over (4). Slight contouring. About equivalent to camera or vtr with poor s/n ratio.	37
6	64	Small improvement over (5). No contouring. About equivalent to average low-band vtr.	43
7	128	Slight improvement over (6). Equivalent to best present cameras and vtr's.	49
8	256	Equivalent to anticipated future cameras, vtr's, and other signal sources.	55

NOTE: Based on actual demonstration by Mark Sanders (Ampex Corp.), "TBC-800 Digital Time-Base Corrector," SMPTE Annual Winter Conference, Jan. 25, 1975.

Picture Elements

Fig. 10-13 shows the result when the scanning beam encounters a thin white bar, equal to the width of one picture element, on a black background. The theoretical ideal response would be a rectangular wave as shown by the dash line. Due to the finite size and round shape of the scanning beam, an *aperture effect* results in a pulse response that is essentially a *sine-squared* shape as shown. For a 4-MHz system, the half-amplitude duration (h.a.d.) of the pulse is 0.250 μs. But note that the single pulse represents two picture elements (black-to-white and white-to-black), so that 0.125 μs represents the time (T) of one picture element.

One picture line is 63.5 μs long, including the blanking interval. The *active* picture line (duration of actual picture information) is the total length of 63.5 μs minus the blanking interval of 11.1 μs, or 52.4 μs. Therefore, there are 52.4/0.125, or approximately 420, picture elements per line for a 4-MHz bandwidth.

As mentioned previously, there are two basic sampling frequencies suggested for digitized video. One is three times the color-subcarrier frequency (10.7 MHz), and the other is four times the color-subcarrier frequency (14.3 MHz). The 10.7-MHz sampling rate is the most commonly used at the time of this writing. At the 10.7-MHz rate, the sampling period is 1/10.7 MHz = 0.093 microsecond, or 93 nanoseconds. Thus, for a single active scan line, there are 52.4/0.093 = 563 samples taken, each sample representing any one of 256 brightness levels (for an 8-bit binary word). Thus, the sampling rate, even for the lowest sampling frequency used, is well above the 420 picture elements per line and provides excellent resolution.

Obviously, if a greater number of samples is taken per second, the possible resolution will be improved. At the 14.3-MHz sampling rate, a sample occurs every 0.07 microsecond, or 70 nanoseconds.

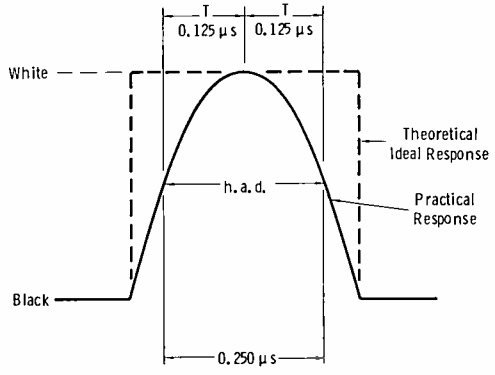


Fig. 10-13. Aperture effect in picture scanning, 4-MHz system.

This means there are $52.4/0.07 = 748$ samples per line. The possible resolving power for the three-times subcarrier rate is down 3 dB at 4.5 MHz. The resolution capability for the four-times rate is down 3 dB at about 6.25 MHz.

However, it is desirable to use the lowest practical sampling rate, since the total transmission rate is the product of the sampling frequency and the number of bits in each sampled level. For the 10.7-MHz rate at eight bits:

$$(10.7\ \text{MHz})(8) = 86\ \text{megabits/second}$$

At the 14.3-MHz sampling rate:

$$(14.3\ \text{MHz})(8) = 115\ \text{megabits/second}$$

Frequency response, or quantizing aperture effect, can be compensated to any desired degree by pre-emphasis before quantizing and de-emphasis following decoding.

Decoding Problems

In the decoding process for pcm digitized tv signals, *comb filters* are often employed. The "teeth" of a luminance comb filter have maximum response at harmonics of the tv line frequency (luminance) and nulls at odd harmonics of half the tv line frequency (color). A color comb filter has teeth with maximum response at odd harmonics of half the tv line frequency (color) and nulls at even harmonics of the tv line frequency (luminance).

When a three-times-color-subcarrier sampling rate is used, an uneven number (three) of samples occurs per quadrature-subcarrier cycle. This presents an offset of alternate time-sequential tv lines to the comb filter, resulting in a certain amount of cross talk between luminance and chrominance, as well as cross talk between the I and Q components.

When a four-times-color-subcarrier sampling rate is used, an even number of samples occurs per I and Q cycle. In this case, samples on all time-sequential tv lines are perfectly aligned, presenting no problem to the comb filters and simplifying the decoding process.

Thus, it would seem that the sampling frequency should be four times the subcarrier frequency. However, a technique known as Phase Alternating Line Encoding (PALE) has been developed to allow digital comb filters to perform about as well at a 10.7-MHz sampling rate as at a 14.3-MHz rate, and with one-third fewer digits. This technique is described in the next subsection.

Phase Alternating Line Encoding (PALE)

In the PALE technique, the phase of the sampling frequency (f_s) is shifted by 180° on alternate horizontal scan lines, as shown

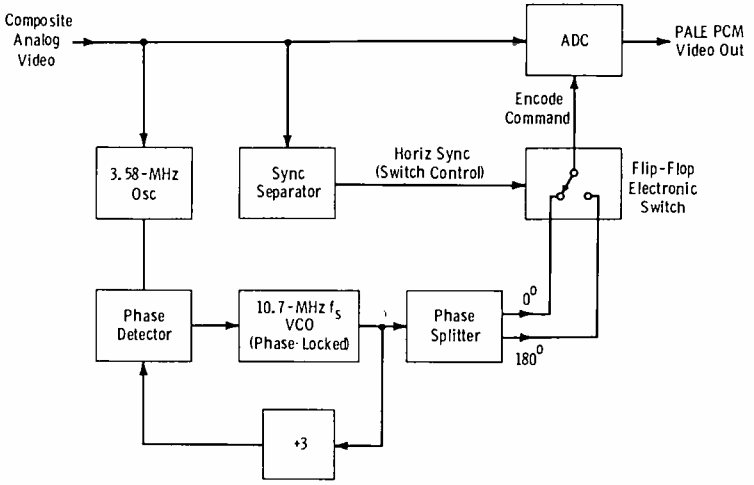


Fig. 10-14. Simplified block diagram of PALE technique.

in Fig. 10-14. The phase shift can be accomplished in the breezeway between the trailing edge of horizontal sync and the leading edge of the color-sync burst.

A means must be provided to identify the phase of the samples to ensure correct signal processing. For example, on one line, a given sample represents +I, but on the next line it represents −I. One system uses sampling angles as follows:

Unshifted lines: 3°, 123° (+1), and 243°
Shifted lines: 63°, 183°, and 303° (−I)

When a signal is stored in a shift register, readout of the stored data requires a reconstructed PALE clock phase-synchronized with the samples on each scan line.

Another problem occurs in the dac output, where switching transients, overshoots, and ringing can occur, particularly if the analog signal has large amplitude differences between adjacent sampling points. One method used to minimize these is termed *resampling* (Fig. 10-15). The dac output is examined by a sampler after it settles in from a previous sample and just before a new number is presented. The output of this sampler is then filtered, which removes practically all of the large glitches. The only detriment is a slight loss of higher frequencies, similar to the aperture scanning loss. Such high-frequency loss can be compensated easily.

Recovery of Color Information

Color information usually is extracted by digital comb filtering and a matrix network. The I and Q color signals are superimposed

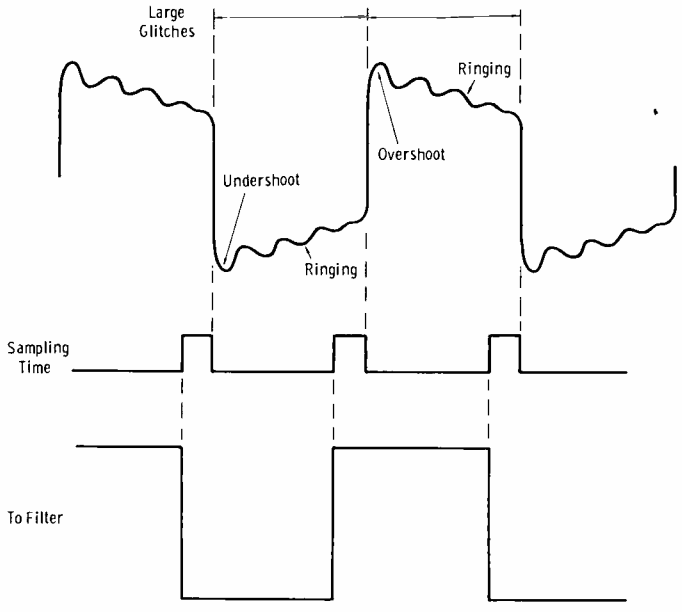


Fig. 10-15. Basic resampling process.

on the dc level of the Y (luminance) component (Fig. 10-16). In Fig. 10-16, the sampling rate is 14.3 MHz. Therefore, the sampling period is 1/14.3 MHz = 70 nanoseconds.

The I and Q signals can be extracted digitally as follows:

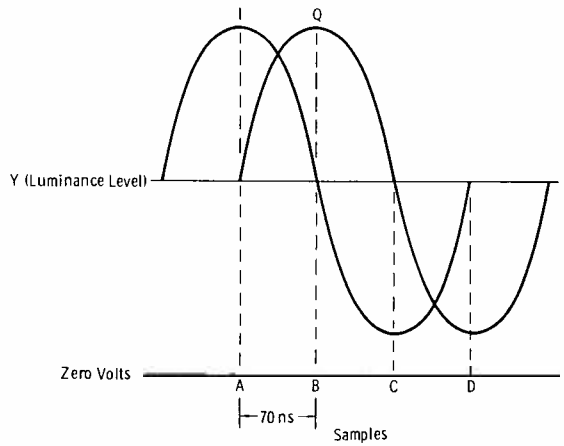


Fig. 10-16. Y, Q, and I components sampled at 14.3-MHz rate.

$$I = A - C$$
$$Q = B - D$$
$$Y = \tfrac{1}{2}(A + C) \text{ or } Y = \tfrac{1}{2}(B + D)$$

This is a highly simplified description. For a rigorous development, see John P. Rossi, "Color Decoding a PCM NTSC Television Signal," *Journal of the SMPTE*, June 1974.

10-5. AUTOMATIC SYNCHRONIZERS

The limitations of genlock (locking the local sync generator to a remote or network signal) have long been realized in tv broadcasting. The entire local system becomes totally dependent on the characteristics of the remote or network signal, which seems a costly trade-off when local mixing and superimpositions over the remote signal may occupy only 5% or less of the total program time. Also, local video-tape systems need rock-solid sync inputs for proper operation, and genlocking normally requires a complete switchable pulse-distribution system to feed an ungenlocked sync generator to production facilities including vtr's.

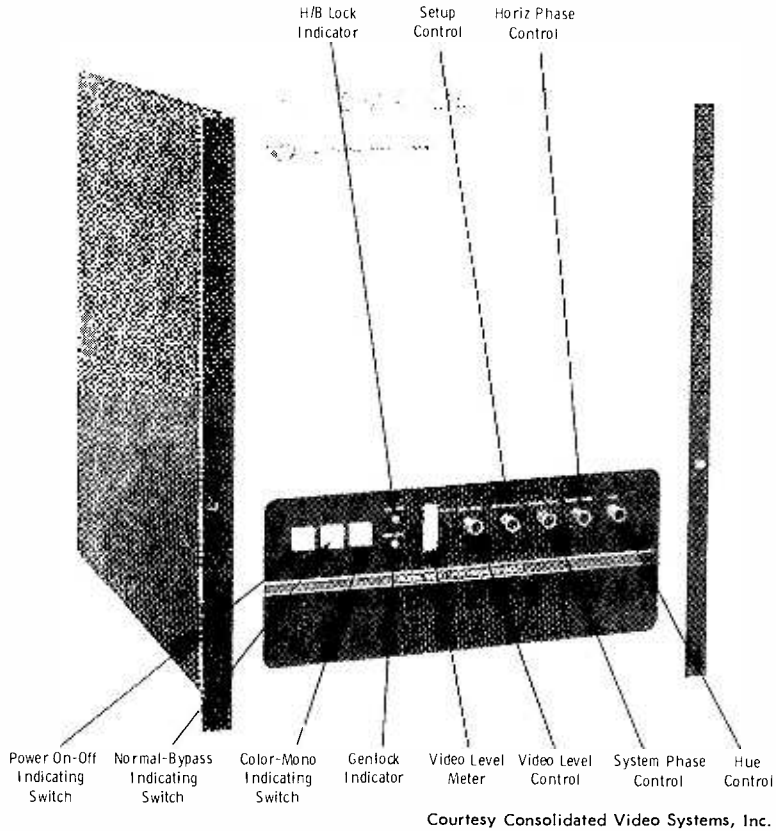
Principle of Automatic Synchronizers

Automatic digital video synchronizers provide "genlock in reverse." The local sync generator remains in its own stable mode of operation without slaving to an external signal. The incoming nonsynchronous signal is processed and converted to digital data, which are stored in a memory bank. The local sync generator then reads the data from the memory bank so that the output is raster and color locked to the local timing information. The data are then restored to analog form and processed for distribution to the local switching system. In this way, any nonsynchronous source(s) can be treated in the switcher just like any of the local sources. Mixing of signals from remote points with local sources presents no problems.

In general, the clocking rate derived from the local sync generator is three times the color-subcarrier frequency, or $3.579545 \times 3 = 10.7$ MHz. The adc forms an 8-bit binary word for each sampled level, therefore "recognizing" any one of 256 video levels. The memory bank in the automatic synchronizer has a capacity of slightly more than one tv field, or approximately 1.5 million bits.

The CVS Model 600 Digital Video Synchronizer

Fig. 10-17 is a front view of the Consolidated Video Systems Model 600 digital video synchronizer. The video level meter on this unit is a vertical panel meter with expanded scale; the center reading is 100 IEEE units, and full scale is ± 40 IEEE units. This meter



Courtesy Consolidated Video Systems, Inc.

Fig. 10-17. Digital video synchronizer.

indicates the absolute value of the input video signal immediately prior to digital conversion, measured from blanking to peak white of the luminance bar on lines 16 through 19 of the vertical-interval test signal.

Most of the controls are self-explanatory. A means is provided to bypass the entire unit in case of failure. In this instance, the remote signal must of course be treated as any other nonsynchronous input signal.

The H/B lock indicator is a green light that is illuminated in the normal monochrome or standby monochrome mode if proper composite sync is present. It also lights in the normal color or standby color mode if proper color reference and composite sync are present.

The CVS-600 is an NTSC television field synchronizer that uses digital storage and processing techniques to synchronize two nonsynchronous television video signals. As shown in the simplified

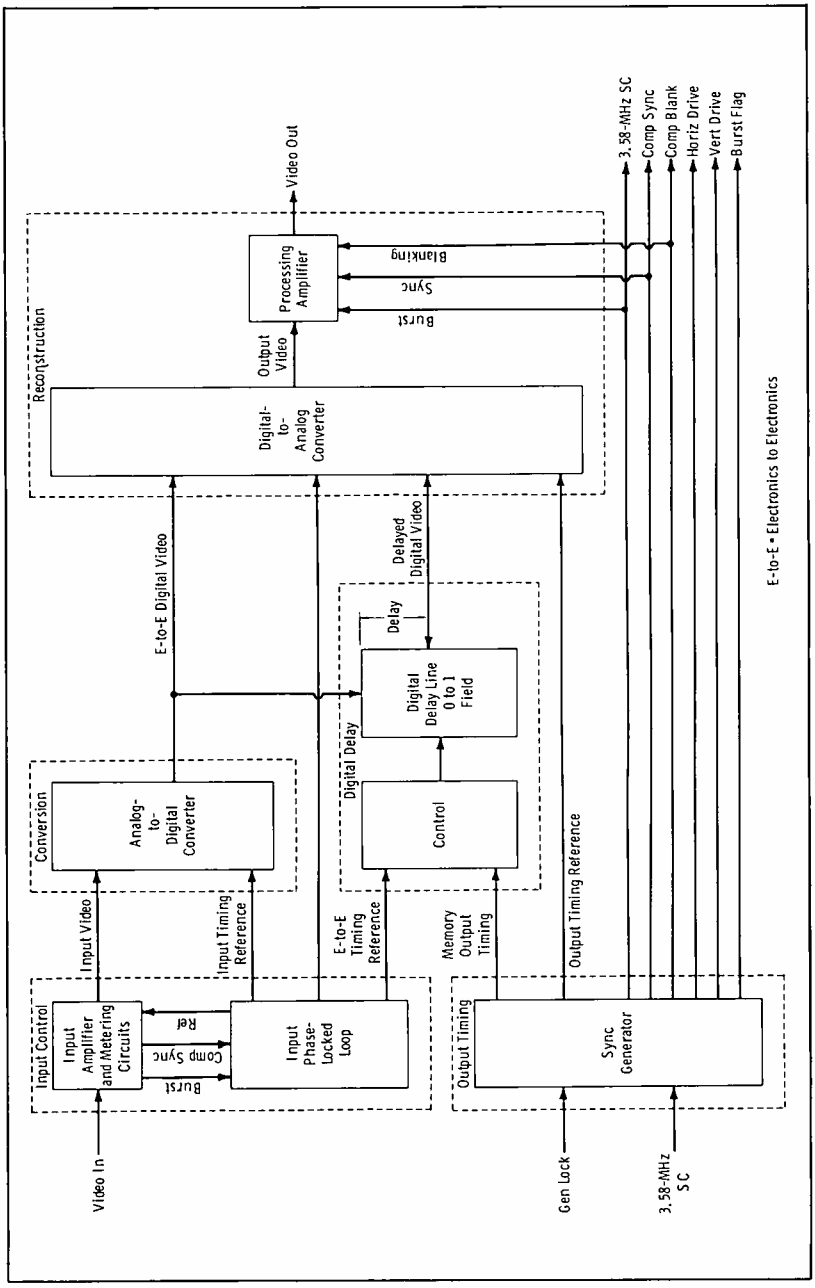


Fig. 10-18. Simplified block diagram of CVS-600 video synchronizer.

block diagram of Fig. 10-18, this is accomplished by five functional circuit groupings: input control, conversion, digital delay, reconstruction, and output timing. The following paragraphs describe these five functional groupings.

The input control circuits consist of the input amplifier and metering circuits and the input phase-locked loop. The input amplifier is of standard design and has provisions for processing a noncomposite input. The amplifier strips sync and burst from the input video and provides these signals to the input phase-locked loop. The video metering circuits provide the means by which the input video is monitored and remotely adjusted. The input phase-locked loop contains a voltage-controlled oscillator (vco) that is phase-locked to the video input by way of the stripped burst and sync signals. (When a monochrome signal is processed, the vco is phase-locked to composite sync only.) The vco output is processed to provide a number of timing reference signals for the other stages of the CVS-600.

The conversion stage consists of an analog-to-digital (a/d) converter. This is a full-bandwidth, 8-bit converter operating at three times the frequency of the color subcarrier (10.7 MHz) if the input is a color signal, or at 682.5 times the horizontal-sync frequency if the input is a monochrome signal.

The digital delay stage aligns the digital video information from the a/d converter stage with a new composite sync signal derived from an internal timing generator in the output timing stage. This timing generator may be phase-locked to external inputs. The delay is continuously and automatically variable from 0 to one full field, and color reference-burst phase and field interlacing are fully preserved. Delay is accomplished by storing the digital video information (without composite sync or blanking) at a rate determined by timing reference signals from the input phase-locked loop and, at the end of the delay period, by providing the digital video information to the reconstruction stage at a rate determined by timing reference signals from the output timing stage.

The reconstruction stage consists of a digital-to-analog converter and a processing amplifier. The 8-bit parallel word from the memory is converted into a noncomposite video signal (sync and burst are not stored) by the d/a converter. The processing amplifier reinserts blanking, sync, and burst, and provides an NTSC signal that meets EIA standards. The signal is then amplified. The result is the automatic synchronization of one NTSC broadcast television signal to another, nonsynchronous NTSC television signal with a lock-up time of 30 milliseconds or less.

The output timing circuits consist of an EIA sync generator and the associated circuits that control the CVS-600 output functions.

While provision is made for locking sync-generator timing to external timing references, the sync generator will free-run when no timing references are applied. Also included in this stage is a vco that generates the output timing references that control the digital delay output functions and reconstruction of the video signal. Sync-generator outputs available at the rear panel are 3.58-MHz subcarrier, composite sync, composite blanking, horizontal drive, vertical drive, and burst flag.

The Analog-to-Digital Converter—The a/d converter is an 8-bit analog-to-digital converter having full video bandwidth and operating at three times the frequency of the color subcarrier. The inputs to the converter are the composite video signal and two timing signals. The output is an 8-bit Gray-code word that is parallel-transferred to the memory bay. Propagation delay for the converter is approximately two clock cycles (186 nanoseconds).

The Digital-to-Analog Converter—The d/a converter (dac) assembly converts 8-bit digital information into cleanly reconstructed and filtered video. The input to the assembly may be the 8-bit digital word produced by the a/d converter or an 8-bit word from the memory bay. When the a/d converter is selected as the data source (E-to-E mode), a timing reference derived from the input video is used to ensure the reconstruction of a properly constituted video waveform. When the memory bay is selected as the data source, the timing reference is derived from the source (output vco assembly) that generates memory output timing signals. The reconstructed signal is then filtered, buffered, and routed to the processing amplifier assembly. The following paragraphs describe the circuits of the d/a converter and are keyed to Fig. 10-19, the block diagram of the assembly.

The 8-bit ECL input from the a/d converter is applied to differential line receivers. The resulting eight TTL signals are applied as one set of inputs to the 8-bit multiplexer. If the multiplexer is properly enabled, the eight bits are passed through the multiplexer and stored in an 8-bit register.

The second set of inputs to the dac assembly is an 8-bit TTL input from the memory bay. This second input is similar to the first; however, the timing relationship with the input video signal is no longer present. The eight signals are applied directly to the input multiplexer. Again, if the multiplexer is so enabled, the eight bits are stored in the 8-bit register.

The multiplexer is controlled by the $\overline{\text{E-to-E}}$ signal. This signal may be applied from the vco divider assembly (from SPLIT SCREEN switch) or generated on the dac assembly by the E TO E switch. The signal is applied to the data selector, which controls the multiplexer and the selection of timing signals. When the $\overline{\text{E-to-E}}$ signal is present

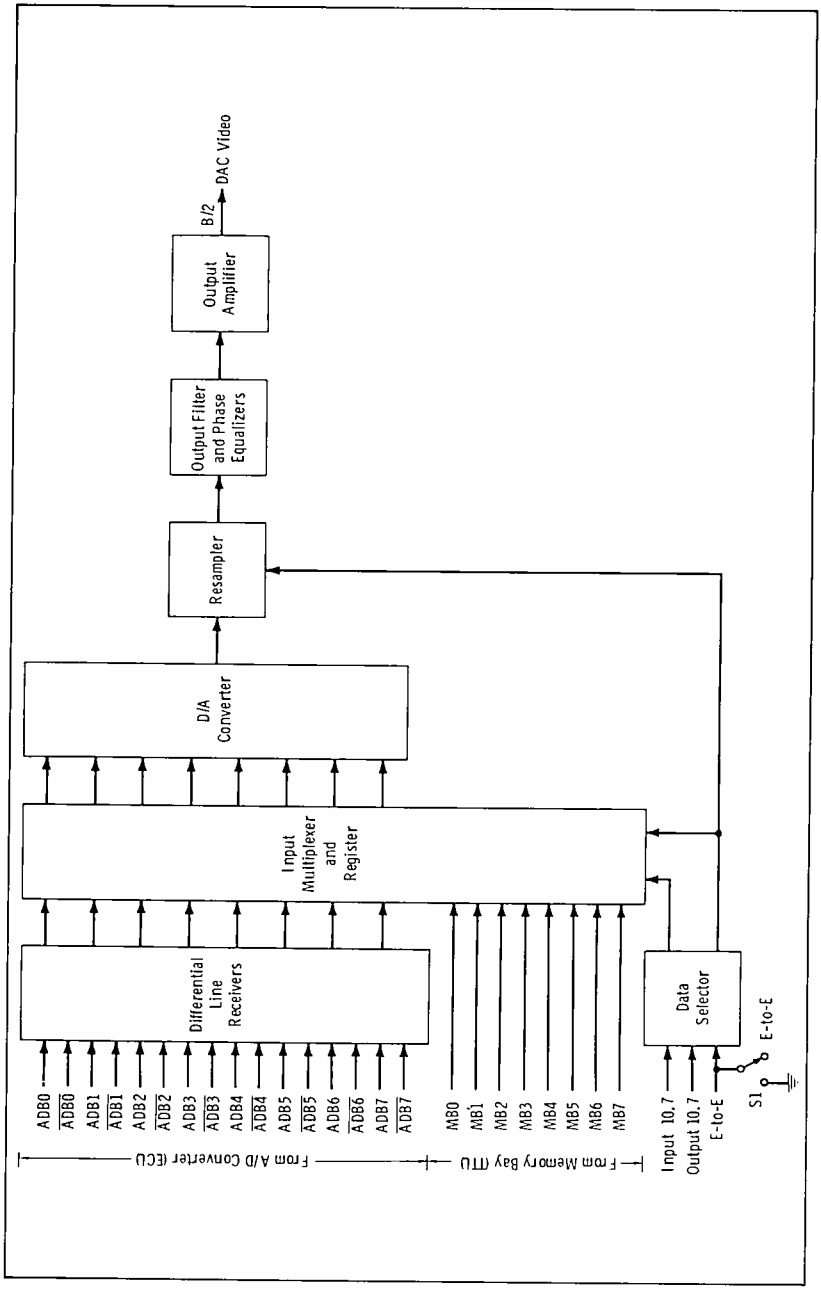


Fig. 10-19. Block diagram of d/a converter assembly.

(low), the multiplexer accepts and passes the input from the differential line receivers. If the signal is high, the 8-bit input from the memory bay is passed to the register.

In addition to controlling the input multiplexer, the data selector passes one of two timing signals to the register and other circuits of the dac assembly. The two signals are input 10.7 and output 10.7. The input 10.7 signal is a 10.7-MHz signal derived from the input vco. The signal has a fixed relationship with the input video and the 8-bit input from the a/d converter. When the $\overline{\text{E-to-E}}$ signal is present (low), the input 10.7 signal controls dac assembly operation. The output 10.7 signal is a 10.7-MHz signal derived from the output vco. This signal is used to generate memory output timing and has a fixed relationship with the 8-bit input from the memory bay. When the $\overline{\text{E-to-E}}$ signal is inactive (high), the output 10.7 signal controls dac assembly operation.

The register consists of eight D-type flip-flops and serves the purpose of deskewing the input. The eight bits from the multiplexer must be applied simultaneously to the dac module to produce a properly reconstructed video waveform. The output of the dac module is applied to a resampler circuit that removes transients introduced by the digital-to-analog conversion. The resampler allows one-half of a clock cycle (approximately 46 nanoseconds) for

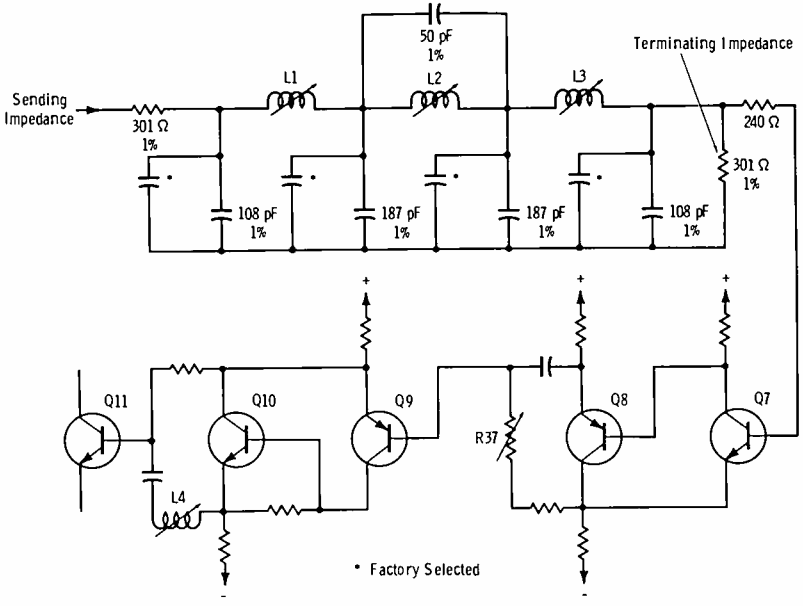


Fig. 10-20. Output video filter and phase equalizers.

the output of the dac module to stabilize. The dac output is then sampled for the remaining 46 nanoseconds, to remove step transients and switching spikes from the video waveform. The reconstructed video is then routed through the output filter, phase equalizers, and output amplifier and applied to the processing-amplifier assembly.

The output filter (Fig. 10-20) attenuates all frequencies above 4.7 MHz, removing the 10.7-MHz sampling frequency and its upper and lower sidebands. Adjustment of this filter is critical and is not attempted in the field. Potentiometer R37 and inductor L4 in the phase equalizers (Q7 through Q11) provide system alignment for proper k factor and group delay.

The output amplifier buffers the output video and drives the output line to the processing-amplifier assembly. The output amplifier has a nominal gain of 5.

Video Compressor Option—Another important feature of digital video is that once stored in memory, the data can be read out on any desired time base. The optional CVS-600-2 video compressor clocks out the video data contained in the memory so that the resultant picture not only is locked to a reference input, but also is reduced 2-to-1 in both the horizontal and vertical directions (equivalent to one-fourth normal raster). That portion of the output signal not containing the compressed video information is blanked, and a key output is provided for key insert. Fig. 10-21 illustrates the capabilities of such an option.

10-6. DIGITAL TIME-BASE CORRECTORS (DTBCs)

Since the introduction of the first practical quadruplex video tape recorder (vtr) in 1956, the need for some form of error correction has existed. Errors inherent in the quadruplex system include scalloping, skew, and quadrature errors.

Forms of Time-Base Correctors

The earlier forms of video-error correctors were voltage-controlled variable delay lines, through which the analog video signal was passed under timing control derived from station sync. For tight control, part of this information was fed back to the headwheel servo for phasing purposes. An accumulated error of greater than about 1 μs often caused instability and failure of the servos to lock up.

With the coming of color, still tighter control of time-base errors was required. Color correction also used voltage-controlled video delay lines, but it compared reference subcarrier to off-tape burst to control the line. While digital as well as analog techniques were used in correction, the system was still primarily an analog control

(analog video controlled) by a combination of digital and analog circuitry. This was an application such as those covered in Chapter 9 (digital control of analog signals).

The digital control of analog vtr signals was not covered in Chapter 9 because such systems are thoroughly covered elsewhere.[5] This text will explore the most recent advances in digital time-base correctors (tbc's), which are becoming very popular.

Part of the reason for the popularity of digital time-base correctors is that modern telecasting employs a large amount of electronic news gathering (ENG), which often requires simple, lightweight helical-scan vtr's for field and mobile use. Until the development of digital time-base correctors, the use of the slant-track (helical) tape recorder was not possible for broadcast purposes. Whereas the accumulated errors of quad tape systems can normally be reduced below one microsecond, helical recorders often require a correction "window" to plus or minus one tv line (63.5 μs), which means that the window has a width of over 125 microseconds.

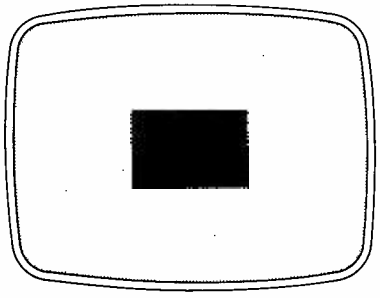
Very briefly, the process is as follows: The demodulated video from the vtr is input-processed in the tbc and converted to a digital coded word. This is loaded into the digital memory at a rate directly related to off-tape sync and burst, which contain all the original time-base errors. The signal is now read out of the memory at a different rate, one that is directly related to the reference sync and subcarrier of the local station or a built-in generator in the tbc.

The Ampex TBC-800 DTBC

The following description[6] of the Ampex TBC-800 digital time-base corrector will serve to illustrate the techniques used in a typical

5. For example, see Harold E. Ennes, *Television Broadcasting: Tape and Disc Recording Systems* (Indianapolis: Howard W. Sams & Co., Inc., 1973).

6. Courtesy Ampex Corp.

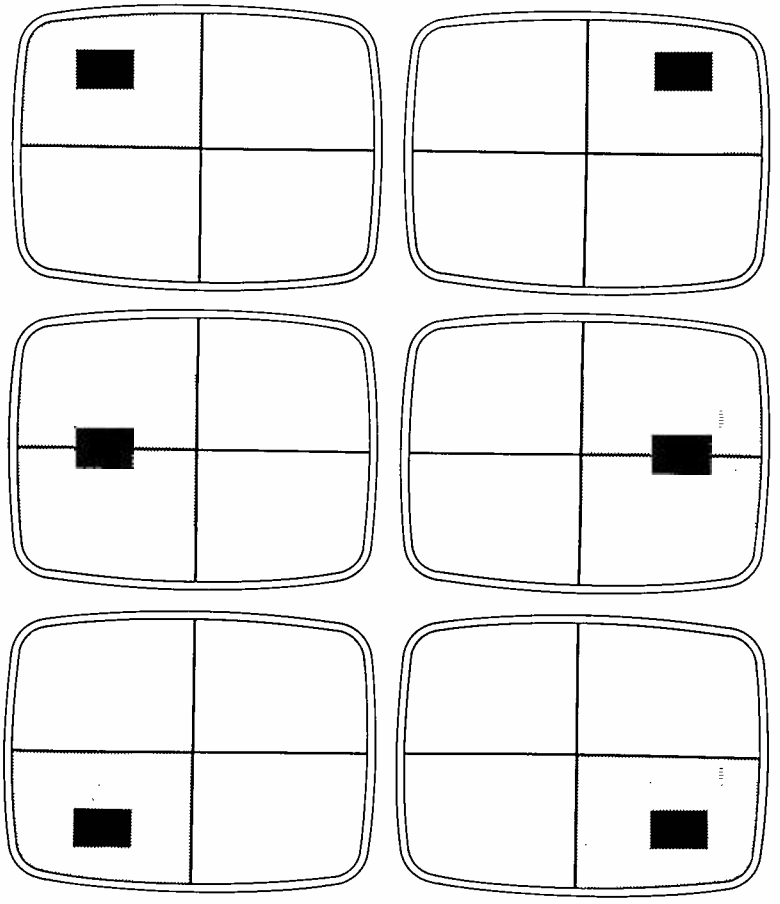


(A) *Incoming signal (synchronous or nonsynchronous).*

Fig. 10-21. Function of

high-quality system. This unit includes a dropout compensator; an optional velocity compensator for color is available.

Input Processing (Fig. 10-22)—Demodulated video is inserted into the input processing block, where conditioning, such as establishing a zero reference, clamping, and filtering, is accomplished. In addition, the burst is amplified to improve the s/n ratio, so the signal exists as slightly modified video. If the signal is monochrome, burst is inserted. Other functions such as sync generation are also shown in Fig. 10-22. The TBC-800 sync generator provides all necessary reference signals and can be genlocked or used as a standard sync source.



(B) Possible positions of output signal (synchronous).

compressed-video option for CVS-600.

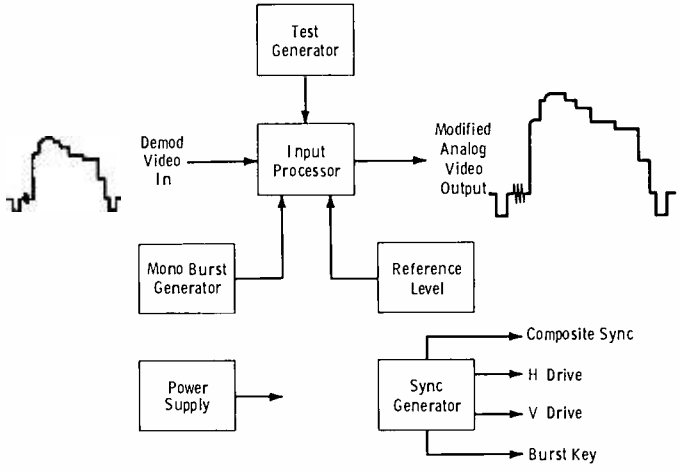


Fig. 10-22. Input processing.

As shown, built-in test signals are incorporated to simplify routine testing of the digital memory and the a/d and d/a converters. The test signals are also available at the outputs of the tbc to facilitate setting deviation of the modulator of a vtr and for checking amplitude linearity of amplifiers, switchers, distribution systems, etc.

Analog-to-Digital Conversion (*Fig. 10-23*)—The analog-to-digital process takes place in a fashion such that the four most significant bits (MSB) are converted first. These are then converted back to analog form in a small d/a converter and subtracted from the incoming video. The a/d conversion process is then repeated to yield the four least significant bits (LSB). (Review Section 7-7.) Of course, all eight bits could be converted at once, but the system used is fast and accurate at a reasonable cost.

The two outputs are combined to provide an 8-bit word representing one analog sample. There are about 683 such samples per line, including the blanking interval. These 8-bit words are developed

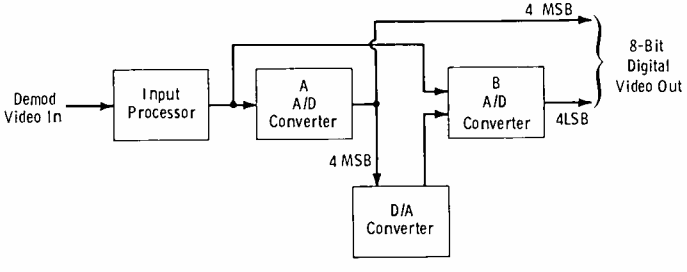


Fig. 10-23. Analog-to-digital conversion.

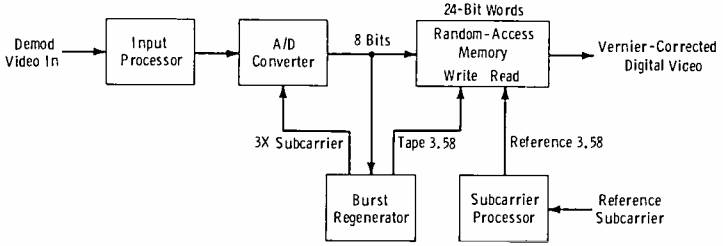


Fig. 10-24. Vernier time-base correction.

at exactly three times the off-tape subcarrier rate, or about 10.7 MHz. They are combined into "triads," or combinations of three words, to make new 24-bit words. These larger words occur at ⅓ the previous rate, or at the off-tape 3.58-MHz subcarrier rate.

Vernier Time-Base Correction (*Fig. 10-24*)—The triads are sampled into a random-access file memory, and in this process the "vernier" time-base correction takes place. Off-tape burst is processed into a small recirculating memory that yields a constant output of off-tape subcarrier. This output contains all time-base error information and is refreshed each line. The burst regenerator is used to time the sampling of the video into the a/d converter and its insertion into memory. At the same time, the reference subcarrier is processed and used to read the data out of memory. Thus, the data are sampled in at the vtr subcarrier rate and are sampled out at the reference subcarrier rate, and phase errors between cycles of the two subcarriers are eliminated. Any further errors are whole multiples of a subcarrier period, or 279 ns. This "vernier" corrected signal is now sent to a much larger random-access memory that is capable of storing more than two horizontal lines of video for "coarse" time-base correction.

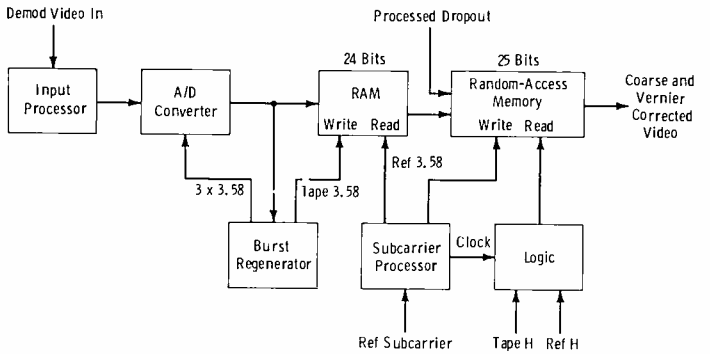


Fig. 10-25. Coarse time-base correction.

Coarse Time-Base Correction (*Fig. 10-25*)—At this point an extra bit is added onto each 24-bit word. This bit contains dropout information obtained from the vtr, and it is combined with the sample of video to which it applies so that it will be delayed the same amount as it passes through the system. The logic circuits compare reference sync to off-tape sync and measure the difference between the two in whole cycles of subcarrier. In other words, the logic looks at a write pulse, counts the number of subcarrier cycles between tape sync and reference sync, and then gives a read command. Thus, errors in increments of cycles of subcarrier are removed, and fully corrected digital video results. The output is now routed to a dynamic shift register (sam, or serial access memory), which provides two lines of delay for dropout reinsertion.

Dropout Compensator (*Fig. 10-26*)—Normally, switch S1 is down, and both the shift control and switch S2 (read) are timed to effect two lines of delay. Thus, one register stores all the odd-numbered lines, and the other stores all the even-numbered lines. If a dropout occurs, the 25th bit will identify it, and just as the bad signal starts to cycle through the register, input switch S1 changes and recirculates the previously stored line of correct phase instead. This may occur at any time, and only the actual dropout will be replaced. (If, however, the dropout occurs during sync or burst, the whole line is replaced, since time-base information has been lost.) The signal, now compensated for dropouts and delayed, is rid of the 25th bit and is converted back to three 8-bit words occurring at three times the subcarrier rate.

D/A Conversion and Velocity Compensator (*Fig. 10-27*)—The digital signal is now routed to the digital-to-analog converter, which translates it back to an analog signal. Due to the nature of the d/a process, however, large switching transients occur at three times the subcarrier rate, and a resampling circuit operating at the same

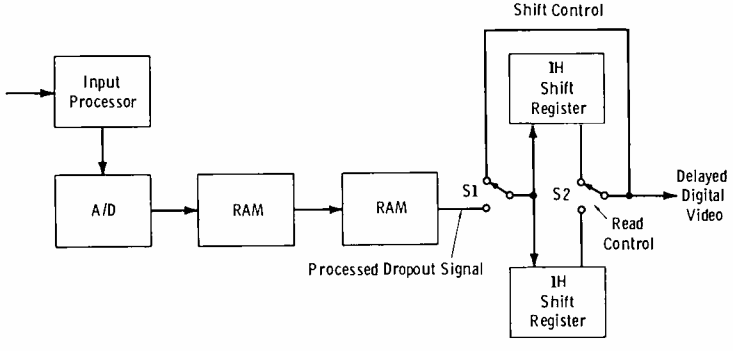


Fig. 10-26. Two-line-delay dropout compensator.

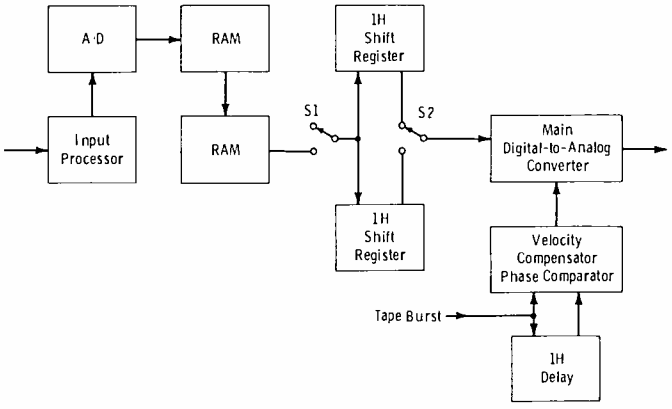


Fig. 10-27. Digital-to-analog conversion and velocity compensator.

10.7-MHz rate is used to remove these transients. Resampling presents an additional opportunity: The phase of the resampling can be altered slightly to provide a means of velocity compensation.

The TBC-800 corrects errors on a line-by-line basis, each time sync and burst information are available. However, residual errors may remain which are caused by the minute mechanical inaccuracies that exist in all recorders. These errors accumulate within the line as it is scanned from left to right and are called *velocity errors.*

For velocity compensation, yet another memory, which delays burst one line, is used. The delayed burst is phased-compared to the burst from the next line, and an error signal representing the difference is developed. This "chroma phase" difference is used to generate a linear ramp, the maximum height of which is proportional to the phase error. This ramp in turn alters the resampling timing in the d/a converter to cancel the effects of velocity errors.

Simplified System (Fig. 10-28)—The video from the d/a converter is routed to the output processor, which filters the signal and removes the resampling effects. The signal is clamped, sync is reinserted along with new blanking and burst, and the time-base correction process is complete.

An optional provision permits color recovery from heterodyne color vtr's as well as direct color systems. With this added provision, the TBC-800 may be interfaced with any capstan-servoed, nonsegmented helical-scan video tape recorder.

Operational Procedures and Adjustments

The operational practices for the TBC-800 described below serve to acquaint the reader with digital tbc's and to indicate the simplicity of such operations. Under normal operating circumstances,

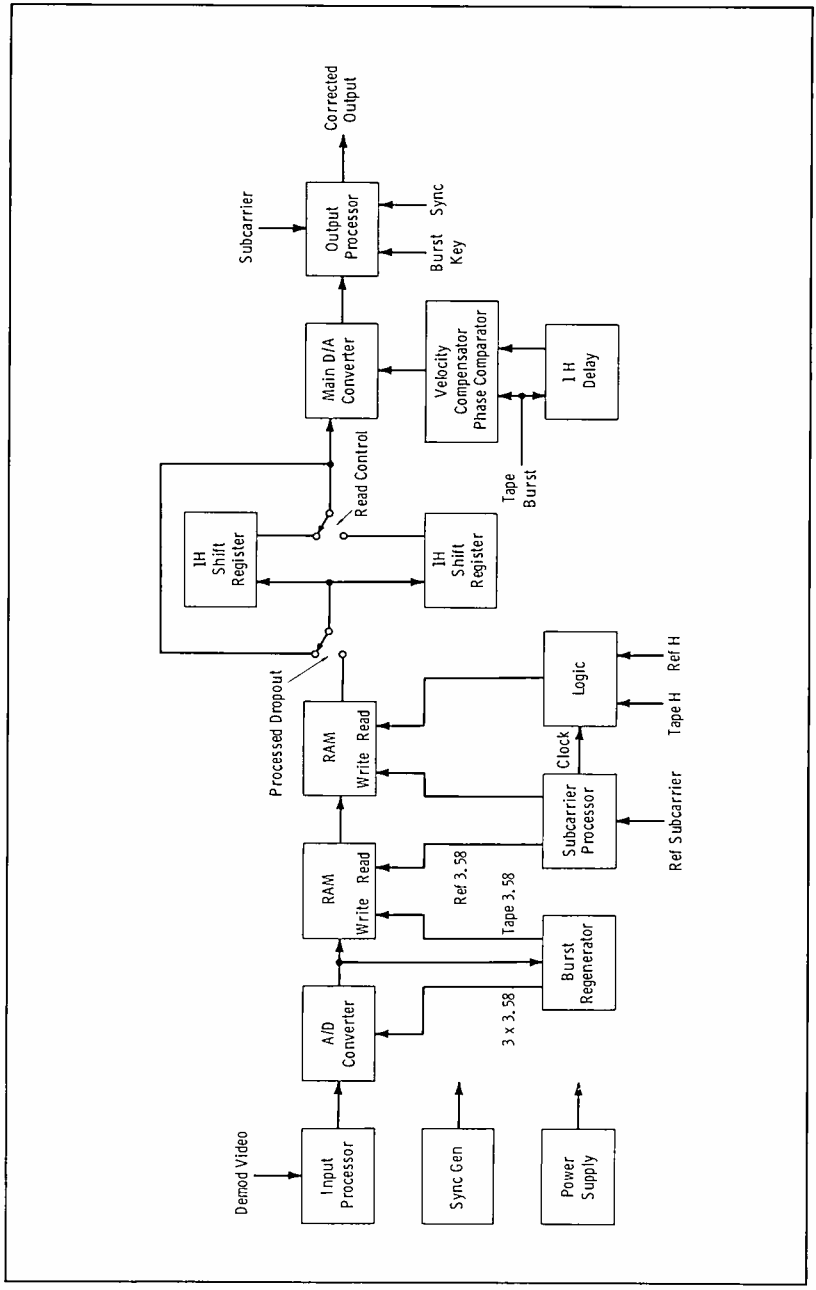
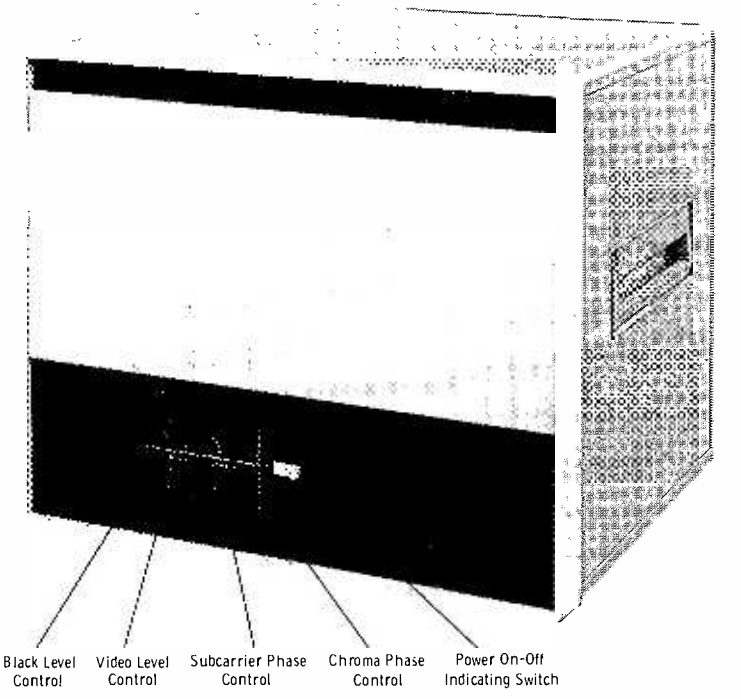


Fig. 10-28. Simplified block diagram of TBC-800.

the controls are adjusted in a preoperational procedure to ensure unity gain and correct phase relationships, and then no further adjustments are required.

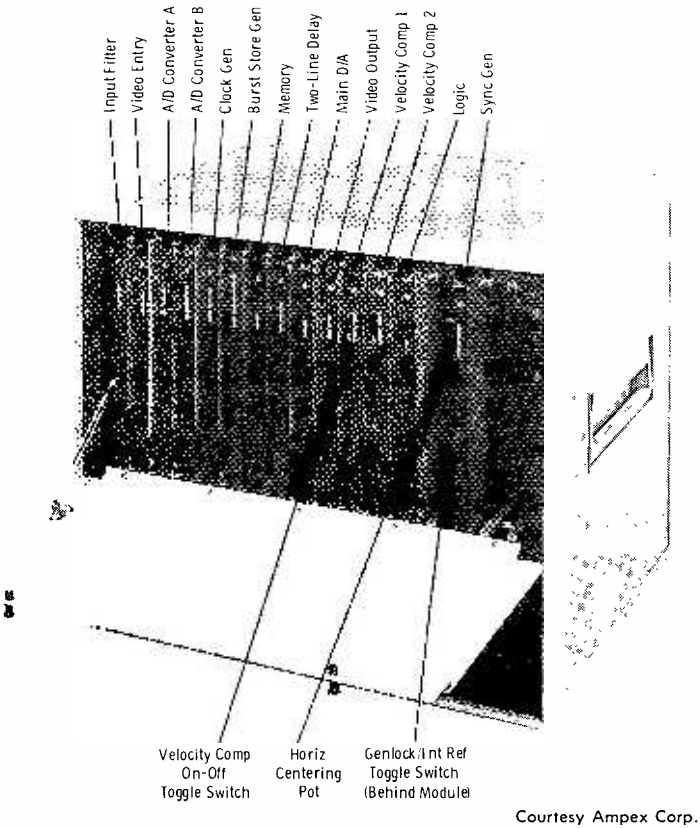
Fig. 10-29 is a front view of the TBC-800 showing operating controls. These have the following functions:

1. Power ON-OFF indicating switch, turns power on and off, and indicates (lights) when switch is in the on position.
2. SUBCARRIER PHASE control sets color subcarrier phase of output signal with respect to external subcarrier reference signal. Used during fully synchronous operation to match color subcarrier phase to external signal sources.
3. VIDEO LEVEL control adjusts the level of the video output signal. Push control to "in" position for unity gain.
4. BLACK LEVEL control adjusts the level between black and blanking level. Push control to "in" position for unity black level.
5. CHROMA PHASE control adjusts the phase of the picture chrominance information with respect to color burst during playback. Push control to "in" position for unity chroma phase.



Courtesy Ampex Corp.

Fig. 10-29. Front view of TBC-800 dtbc.



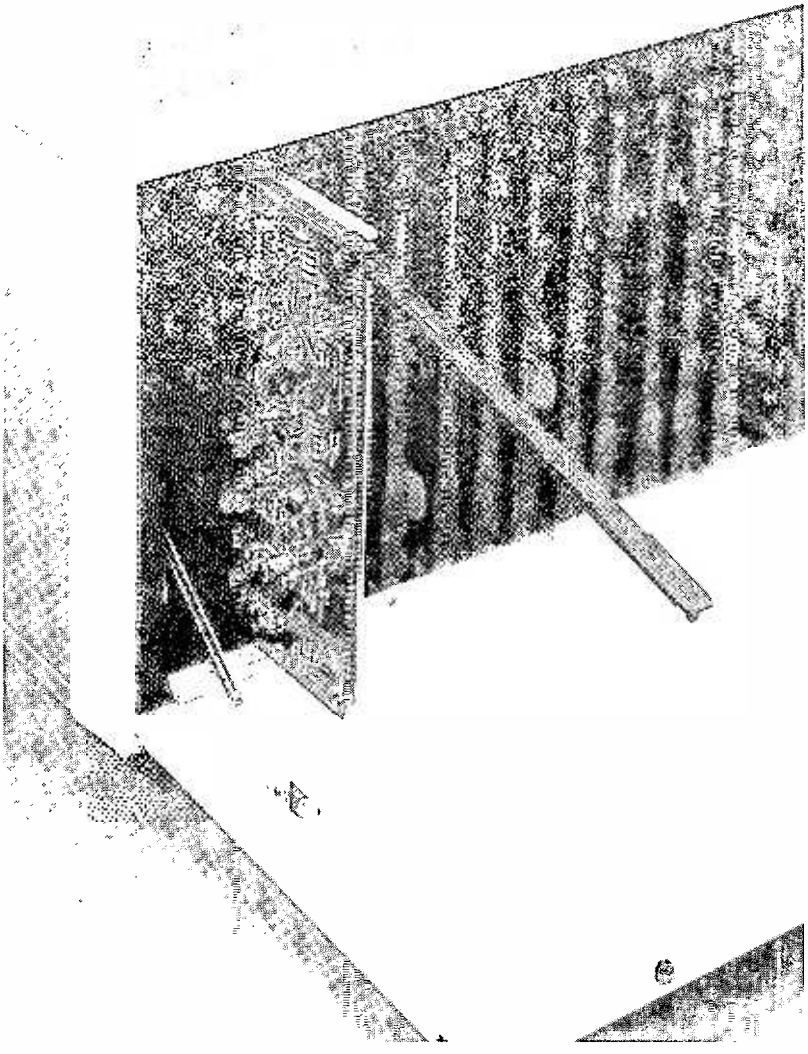
Courtesy Ampex Corp.

Fig. 10-30. Identification of modules in TBC-800.

Fig. 10-30 is a view of the TBC-800 with the top cover opened to reveal the control modules. The three adjustments provided on the control modules are indicated. Fig. 10-31 shows the accessibility of the plug-in modules.

Preoperational Procedure—After the TBC-800 has been installed and connected to the associated recorder, the following preoperational procedure may be performed. All controls are accessible from the front of the unit.

1. Set the GEN LOCK/INT REF toggle switch as follows:
   A. Nonsynchronous installation, Int Ref (down position).
   B. Synchronous installation, Gen Lock (up position).
2. Connect a 1-volt p-p color-bar signal to the input of the recorder. Apply power to the recorder and the TBC-800 and place recorder in standby (stop) mode.



Courtesy Ampex Corp.

Fig. 10-31. Module arrangement for servicing.

3. Connect a waveform monitor or oscilloscope to the video output connector of the recorder. Use a 75-ohm termination at the input to the monitor or scope. Verify that the signal is 1-volt p-p (unity gain).
4. Record and play back a one-minute segment of the color-bar signal. Signal during playback should have a 1-volt p-p amplitude with correct chroma levels. If signal is not correct, adjust the recorder as necessary.
5. Connect a waveform monitor or scope to one of the unused TBC-800 video outputs. Use a 75-ohm termination at the input to the monitor or scope. Check to see that the burst and sync levels are 0.286 volt p-p (40 IEEE units). Be sure that 140 IEEE units = 1 volt.

Video and Black-Level Adjustment—Check the video and black levels, and if they are correct, leave the VIDEO LEVEL and BLACK LEVEL potentiometer/switches in the "in" (unity) position. Check the levels as follows:

1. Record and play back a one-minute segment of color-bar signal.
2. While playing back the tape, check to see that the video level and black level are 0.714 volt p-p (100 IEEE units). Normally, black level (setup) is from 0 to 10%, depending on the system requirements.
3. If the levels are incorrect, remove the controls from the unity ("in") position and adjust as necessary.

Chroma Phase Adjustment—If a tape being played back has been recorded with the correct burst-to-chroma phase, place the CHROMA PHASE control on the TBC-800 in the unity ("in") position. If the tape has been recorded with incorrect burst-to-chroma phase, the CHROMA PHASE control on the TBC-800 may be used to correct the error. The CHROMA PHASE control can shift the burst-to-chroma phase over a range of ±30°. To adjust chroma phase, proceed as follows:

1. Connect a vectorscope to one of the unused TBC-800 video outputs. Use a 75-ohm termination at the input of the vectorscope.
2. While playing back the tape to be corrected, adjust the vectorscope phase control to place the burst vector at the correct point on the polar display.
3. Adjust the CHROMA PHASE control on the TBC-800 to place the color vectors at the correct points on the polar display.

System Subcarrier Phase Adjustment—To permit synchronous operation of the recorder and the TBC-800 system with another color

source, the subcarrier phase of the TBC-800 output signal may be adjusted to match the phase of the external signal as follows:

1. Connect a dual-trace scope to the video output of the TBC-800 and to the video output of the external signal source. Use 75-ohm terminations at the input of the scope.
2. Adjust the scope to display one horizontal line.
3. Record and play back a one-minute segment of tape. While playing back the tape, adjust the horizontal centering potentiometer on the sync-generator module to line up the leading edges of the two sync signals.
4. Disconnect the scope.
5. Connect a dual-trace vectorscope to the video output of the external signal source and the video output of the TBC-800. Use 75-ohm terminations at the input of the vectorscope.
6. Record and play back a color-bar segment of tape. While playing back the tape, adjust the vectorscope for dual display and adjust the SUBCARRIER PHASE control on the TBC-800 to superimpose the bursts of both signals.

Routine Operation—If standard-level input signals are supplied to the TBC-800, the VIDEO LEVEL, RLACK LEVEL, and CHROMA PHASE controls may be left in their unity ("in") position. If the input levels are not standard, adjust these controls as described in the preoperational procedure.

To obtain maximum performance from interchanged tapes and to check system levels, it is suggested that a one-minute segment of color bars be recorded at the beginning of each tape. The recorded segment provides a reference that the operator can use later to check levels and phase. If required, adjust the controls to provide an output signal with standard levels.

10-7. ELECTRONIC STILL STORE (ESS)

In present-day telecasting, video stills depend primarily on either photographic 35-mm transparencies or opaque graphic material. By far the largest percentage of stills originate from slide projectors; the average tv station maintains a file of 2000 to 5000 slides. This presents a mammoth filing problem in addition to the laborious process of photographing, developing, and mounting the individual transparencies.

In mid-January of 1976, the Ampex Corporation and the Columbia Broadcasting System announced the joint development of an all-electronic "slide projector." This system, termed *Electronic Still Storage (ESS)*, uses digital magnetic recording on multiple-surfaced computer disc packs. (Review Section 6-4.)

Ten discs are contained in each pack, with a recording surface and a head on each side of each disc. The ten discs provide twenty surfaces, numbered from surface 0 to surface 19. Each pack stores up to 1472 stills. The following description of the Ampex ESS[7] will provide the reader with some insight into the theory of operation.

Digital Video Signal System

General Considerations—With the prerequisite that heads and discs remain unmodified, we need to examine the signal performance of the computer disc drive compared with the needs of video recording. The Ampex DM-331 drive is a machine equivalent to, and plug-compatible with, an IBM 3331 drive. This drive model was chosen primarily because of its large bulk-storage capacity of 200 megabytes (1 byte = 8 bits) per disc pack and its convenient rotational speed of 3600 rpm. Due to the wide separation of heads and discs, the throughput frequency response of a disc read/write channel at 2000 in/s is much narrower than that of a conventional quadruplex vtr at 1500 in/s (product of head circumference and rotational velocity). It is insufficient to accommodate a high-band fm signal and could handle video signals of limited bandwidth only, such as those in a color-under system. Therefore, analog video recording was ruled out since it would have implied a serious sacrifice of picture quality.

The throughput response of the system is capable, however, of accommodating a digital data stream at a rate of 10.7 megabits/second (3 times the subcarrier sampling rate) corresponding to a linear packing density of approximately 6000 bits/inch. This density is approximately 60% above the level of the basic drive and was accomplished by a redesign of the read and write circuits. It corresponds to a bulk-storage capacity of 300 megabytes per pack. The rotational disc speed of 3600 rpm is maintained. Instead of a free-running spindle drive, however, a servo system was designed to lock the pack rotation to the vertical sync. The availability of multiple parallel heads and discs leads directly to the recording of digital video signals on eight parallel tracks at a sampling rate of 10.7 megasamples/second. Recording of one tv field requires one revolution of the disc pack and simultaneous use of eight heads (Fig. 10-32). Recording of a full tv frame requires two revolutions and the use of 16 parallel heads. A disc pack carries 800 cylinders[8] with 19 data tracks each and can therefore store 800 digital tv frames, leaving three tracks for other uses.

7. Courtesy Ampex Corp.
8. A "cylinder" is all tracks on the same radius of a disc pack.

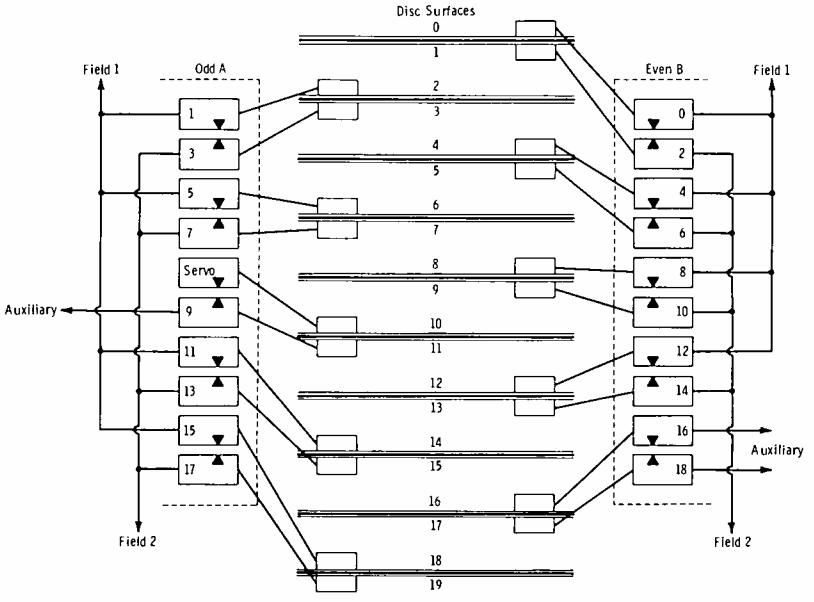


Fig. 10-32. ESS disc-drive head assignments.

The signal-to-noise ratio of this digital tv system is primarily determined by quantization noise rather than by tape and preamp noise as in the case of a vtr. This is true provided that the digital random error rate of the storage channels is low enough to make occasional transmission errors invisible. With quantization as the limiting factor, the system delivers a signal-to-noise ratio of 58 dB. Effects known in vtr's, such as moire, headbanding, and residual time-base error, do not exist.

Block Diagram—In the simplified record-signal flow diagram (Fig. 10-33), the composite video input signal passes through the entry stage where clamping takes place and sync and subcarrier components are stripped and regenerated. The following a/d converter uses components also found in the Ampex TBC-800 digital time-base corrector. At the output of the a/d converter, digital video is available in a parallel 8-bit format sampled at 3 times the subcarrier frequency. Sampling involves use of the PALE method, an inversion of sampling phase every tv line to facilitate digital chroma processing during reproduction. (Review Section 10-4.)

In the next stage, the 8-bit NRZ video word is converted to a special recording code particularly suitable for digital magnetic recording. This code minimizes the dc content of data streams if long strings of similar data are encountered. In addition, each of the eight data lines receives a synchronizing word once every tv line,

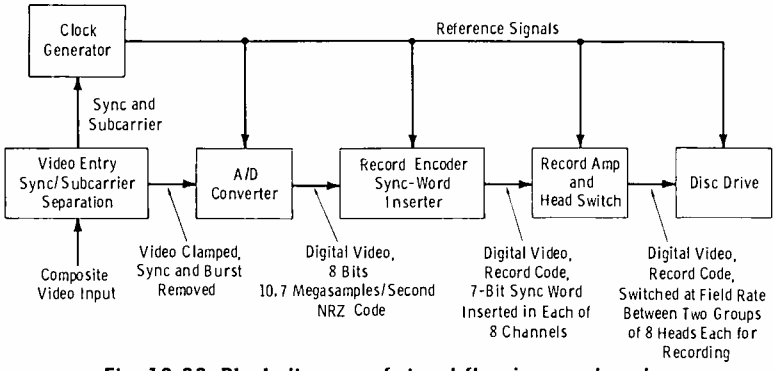


Fig. 10-33. Block diagram of signal flow in record mode.

timed precisely with respect to horizontal sync. On playback, the sync word is used as a reference for correction of time-base and skew error.

To record a full tv frame, the disc performs two revolutions. The first field is directed to one group of eight heads, and the second field to another group of eight heads, so that recording of a frame involves 16 heads without mechanical motion of the head carriage. A total of 19 heads are available in the drive for data recording.

During playback (Fig. 10-34), the signal is fed from two groups of eight heads each into eight preamplifiers, analogous to the two-field record cycle. The following equalizer compensates for the head response and corrects the timing of signal transitions. A zero-crossing detector produces a square wave from the equalizer output, representing the encoded original signal.

In the next stage of the diagram, the signal is decoded to an NRZ format. Each of the eight individual data lines is brought to a time-base corrector that will locate the inserted sync word and use it to remove any head skew and time-base error contained in the signal. The signal is now under station reference. In this manner, all eight data lines are individually corrected, and the original digital video word stream is reproduced.

Before the digital signal can be reconverted to the analog domain, another processing step takes place. In order to regenerate the appropriate four-field NTSC chroma cycle from two recorded fields, the color signal must be separated, inverted at frame rate, and recombined with the luminance. This is accomplished in a comb filter and chroma inverter. As are all the other signal steps in the ESS system, this function is entirely digital. It is simplified by use of the 3-times-subcarrier PALE encoding method mentioned earlier.

A processing amplifier is the final stage of the diagram. Here, sync and subcarrier are reinserted.

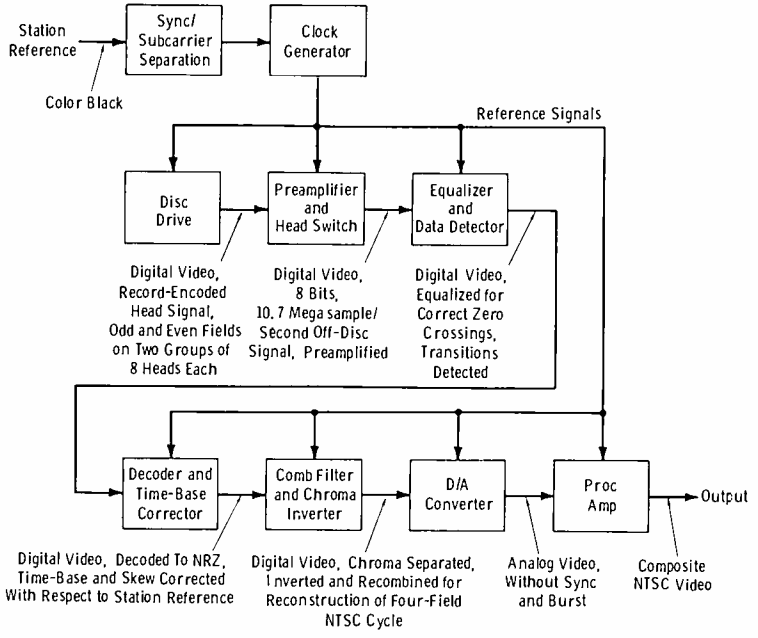


Fig. 10-34. Block diagram of signal flow in playback mode.

Control System

The Electronic Still Store makes use of two disc drives and can be extended to include a third drive (Fig. 10-35). Two drives as a basic system were chosen for the following reasons:

1. For most stations, 1472 stills (736 per drive) cover the current file. Sixty-four stills per drive are reserved as working tracks. Archival material is stored in off-line disc packs.
2. Two drives are mandatory for transfer from archival to on-line disc packs.
3. Preview and vertical-interval switching can be provided.
4. Simultaneous access by at least two users is provided, and the system can be treated similarly to two independent telecines.
5. The redundancy of drives and output channels reduces the probability of a complete system breakdown.

Recording of incoming material on any of the two or three drives is accomplished through a single record channel, with the assumption that simultaneous recording on more than one drive will not be required. Playback occurs through at least two reproduce channels, A and B, that can accept video from any of the drives through a reproduce switcher. Simultaneous playback of two stills is possible

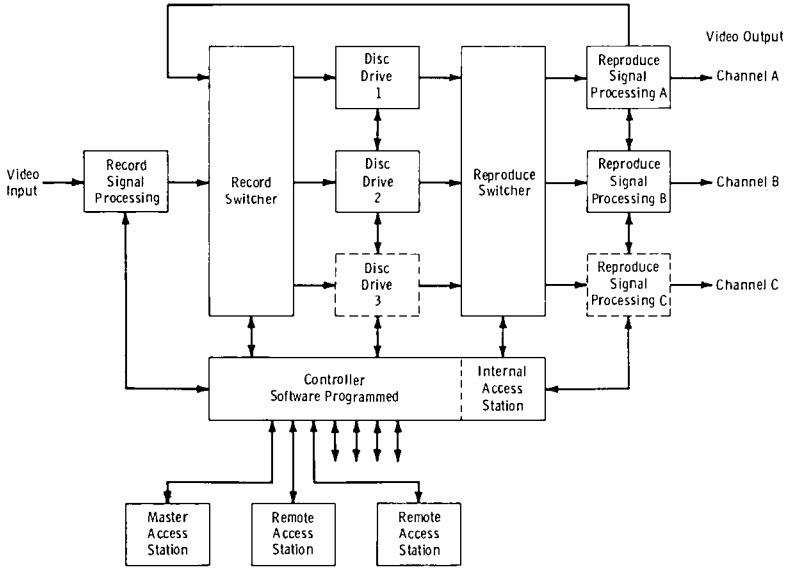


Fig. 10-35. General block diagram of ESS system.

provided they originate in different drives. All reproduce channels accept a common color black reference signal but provide independent output timing and chroma adjustments. Transfer of stills between packs as needed in the sequence assembly described below is accomplished by means of a transfer line from video channel A to the record switcher. This process is totally digital. Time-base and skew errors are corrected during transfer in order to avoid error accumulation.

The ESS system can be controlled from up to eight remote access stations. Access to each disc by several users can be on a first-come, first-served basis, or it can be exclusively assigned, such as during air time. A remote access station consists of a keyboard and a 32-digit alphanumeric display. On the left-hand side of the panel, an operating mode such as Play, Record/Delete, or Sequence Assembly can be selected. Data-entry keys on the right-hand side are used to select channels or individual stills and to manipulate an internal memory used for storage of play sequences. An INITIATE bar at the bottom is depressed following data entry to request execution of the command from the central controller.

Data entry into the keyboard is reflected in the display and can be cleared before execution is requested. After the INITIATE bar has been depressed, the display will serve as the feedback link informing the operator of the central controller response. This response

may consist of numerical address information for video placed on line, or it may be status information relating to control steps being taken.

Major Operating Modes—There are four major operating modes. These are Play, Record/Delete, Sequence Assembly, and Sequence Play. In the Play mode, the operator can instantly access any one of 1472 on-line stills by entering an address and requesting access. While browsing, he may also compose still sequences by transferring desired addresses into an internal sequence memory. This memory can accept a total of 128 still addresses to form one or several independent picture sequences. Video frames representing the stored addresses in the memory can be assembled on reserved working tracks in a separate operation, described below.

The Record/Delete mode is used for recording incoming video stills and for deleting obsolete material in order to clear storage tracks for new recordings. Recordings can be made on specifically addressed tracks, provided these tracks do not contain valid picture material. A code recorded along with any new still will prevent overwriting by another still. Recordings can also be made by asking the controller to search for the next available track.

Obsolete stills in the store can be deleted by means of a key switch on the remote access station. The delete process over-records a special bar pattern and changes the status code from "occupied" to "available." In a typical broadcast plant, the key-operated delete function would be available only on the master access station.

Sequence Assembly is an internal routine to place video signals corresponding to programmed address sequences into special working tracks not used for bulk storage. It involves video transfers between the two on-line disc packs or from off-line packs that have to be loaded on one of the drives. The end result of the assembly process is a translation of still addresses that may involve pictures on several off-line disc packs into sequential video stills recorded on the reserved working tracks of one or two on-line disc packs.

The ability to play a prearranged sequence of stills by means of a simple advance control is the final objective of the sequence list and assembly steps described above. In the Sequence Play mode, the ESS system can be operated from the control room during air time much like two telecines. Two totally independent still sequences on two disc packs are available to the operator and can be advanced by means of two control buttons. Preview and dissolve or wipe effects between stills are possible. In the controller program, the end of each preprogrammed sequence is tied back to its beginning. This enables the technical director to step through the entire sequence for preview purposes and to return automatically to the first still after the last.

Central Controller—The complexity of the control functions and the amount of data to be manipulated are such that only a software system can handle the problem efficiently. A DEC LSI-11 microcomputer was chosen for the task. This computer is a 16-bit machine that combines central processor, 4K of memory, and a buffered 16-bit parallel I/O bus on a single 8-inch × 5-inch circuit card. Additional memory and interface devices are placed on separate circuit cards. The LSI-11 is intended to emulate the PDP-11-40 system. This system, aided by peripherals, was used to develop control software to be applied by the much smaller LSI-11 system in the dedicated ESS environment. In this manner, program development can take advantage of the power and flexibility of a full-size PDP-11, while the actual system control tasks are handled by a small sister machine. The control program of approximately 3000 instructions resides in the main memory and is protected against power failures of up to one-hour duration. The program can be reloaded into the memory from a fast paper-tape reader.

Automated Television Plant—The Electronic Still Store is particularly adaptable to the automated television plant. With the central ESS controller based on a software system, the interface with a host computer will connect two similar worlds. All ESS operating modes and entries presently available to the operator at a remote access station will also be available to the station computer through an interface. This interface will replace one of the eight remote access stations that can presently be accommodated by the ESS controller. The look-up table relating still accession numbers with location addresses in the still store is considered to be a library routine of the automation system rather than an ESS function. In cases where this function will require excessive station computer memory, sufficient data space can be provided in the disc pack to record accession numbers along with location addresses.

Summary

The storage, handling, and broadcasting of stills in today's television plant is a costly manual operation unsuitable for automated systems. In a joint program, Ampex and CBS have developed a totally digital Electronic Still Store based on a computer disc drive as the central storage element. The system provides instant random access to 1472 stills and can store additional stills off-line. Still sequences can be composed and assembled for playback at air time. The emphasis of the design is on reliability, picture quality, and flexibility of multiple access control and interface with a station automation system.

The design of the access system provides stack acceleration sufficient to traverse the entire range of 800 tracks in approximately 60

milliseconds, including settling, and to move between adjacent tracks in 10 milliseconds. This time should be compared to the 200 milliseconds required for a telecine mirror flip.

BIBLIOGRAPHY

Acker, David E. and McLean, Richard H. "Digital Time-Base Correction for Video Signal Processing." *SMPTE Journal*, March 1976.

Bennett, W. R., "Spectra of Quantized Signals." *Bell System Tech. Journal*, July 1948.

Bhushan, A. K. "Experiments in Coding Color Pictures." *Proceedings of the Symposium on Picture Bandwidth Compression*, Massachusetts Institute of Technology, April 1969.

Busby, E. S., Jr. "Principles of Digital Television Simplified." *SMPTE Journal*, July 1975.

Cattermole, K. W. *Principles of Pulse Code Modulation*. New York: American Elsevier Publishing Co., 1969.

Chambers, J. P. *The Use of Coding Techniques to Reduce the Tape Consumption of Digital Television Recordings.*" Paper presented October 1973 at SMPTE 114th Technical Conference in New York.

Chambers, J. P. and Jones, A. H. *Digital Magnetic Recording: Conventional Saturation Techniques*. BBC Research Dept., Report No. 1972/9.

Cutler, C. C. *Differential Quantization of Communication Signals*. U.S. Patent 2,605,361, July 29, 1952.

Davidoff, Frank. "Digital Video Recording for Television Broadcasting." *SMPTE Journal*, July 1975.

———. "Technical Note: IEEE Trial-Use Standard 511-1974 on the Measurement of Linear Waveform Distortion in Video Signal Transmission." *SMPTE Journal*, September 1975.

Devereux, V. "Applications of PCM to Broadcast Quality Video Signals." *The Radio and Electronic Engineer*, September 1974.

———. *Bit-Rate Reduction of Digital Video Signals Using Differential PCM Techniques*. International Broadcast Conference, London, September 1974.

———. "Pulse Code Modulation of Video Signals: Subjective Study of Coding Parameters." *The Radio and Electronic Engineer*, July 1974.

Dorwood, R. M. "Aspects of the Quantization Noise Associated With the Digital Coding of Color-Television Signals." *Electronic Letters*, January 1970.

Felix, Michael O. "Differential Phase and Gain Measurements in Digitized Video Signals." *SMPTE Journal*, February 1976.

Fink, D. *Electronics Engineers Handbook.* New York: McGraw-Hill Book Co., 1975.

Goddall, W. M. "Television by Pulse Code Modulation." *Bell System Tech. Journal,* January 1951.

Gold, Bernard, and Rader, Charles M. *Digital Processing of Signals.* New York: McGraw-Hill Book Co., 1969.

Goldberg, A. A. "PCM Encoded NTSC Color Television Subjective Tests." *SMPTE Journal,* August 1973.

———. "PCM NTSC Television Characteristics." *SMPTE Journal,* March 1976.

Golding, Leonard S., and Garlow, Ronald K. "Frequency Interleave Sampling of a Color Television Signal." *IEEE Trans. Communication Technology,* December 1971.

Griffin, J. S. *Ultra High Data Rate Digital Recording.* Paper presented at the SPIE Conference on Military Airborne Video Recording in Dayton, Ohio, April 1973.

Howell, David A. "A Primer on Digital Television." *SMPTE Journal,* July 1975.

Jones, A. H. *Digital Television Recording: A Review of Current Development.* BBC Research Dept. Report No. 1973/29.

Jones, A. H., and Bellis, F. A. *An Experimental Approach to Digital Television Recording.* International Broadcast Conference, London, September 1973.

Limb, J. O. "Design of Dither Waveforms for Quantized Visual Signals." *Bell System Tech. Journal,* September 1969.

Limb, J. O., and Rubinstein, C. B. "Plateau Coding of the Chrominance Component of Color Picture Signals." *IEEE Trans. Communications,* June 1974.

Limb, J. O., Rubinstein, C. B., and Walsh, K. A. "Digital Coding of Color Picturephone® Signals by Element Differential Quantization." *IEEE Trans. Communication Technology,* December 1971.

Maegele, Manfred. "Digital Transmission of Two Television Sound Channels in Horizontal Blanking." *SMPTE Journal,* February 1975.

Nyquist, H. "Certain Factors Affecting Telegraph Speed." *Bell System Tech. Journal,* March 1924.

Rabiner, Lawrence R., and Rader, Charles M., Editors. *Digital Signal Processing.* New York: IEEE Press, 1972.

Roberts, L. G. "Picture Coding Using Pseudo-Random Noise." *IEEE Trans. Information Theory,* February 1962.

Rossi, John P. "Color Decoding a PCM NTSC Television Signal." *SMPTE Journal,* June 1974.

———. "Digital Television Image Enhancement." *SMPTE Journal*, July 1975.

———. "Sub-Nyquist-Encoded PCM NTSC Color Television." *SMPTE Journal*, January 1976.

Rubinstein, C. B., and Limb, J. O. "Statistical Dependence Between Components of a Differentially Quantized Color Signal." *IEEE Trans. Communications*, October 1972.

Spitzer, Charles F. *Digital Recording of Video Signals up to 50 MHz.* Paper presented at the SPIE Conference on Military Airborne Video Recording in Dayton, Ohio, April 1973.

U. S. Patent No. 3,946,432.

Wetmore, R. Evans. "DATE: A Digital Audio System for Television." *SMPTE Journal*, March 1974.

EXERCISES

Q10-1. Give two basic reasons for converting analog signals to digital form.

Q10-2. Why should an analog signal ever need temporary storage in digital form?

Q10-3. Why are pre-emphasis and de-emphasis normally used in digital audio and video systems?

Q10-4. Why is a low-pass filter always used between an analog signal and the adc?

Q10-5. Why is a low-pass filter always used between the dac and the analog output?

Q10-6. What is the purpose of a "resampler" following a dac?

Q10-7. What determines the minimum sampling frequency?

Q10-8. Relative to the analog signal input level, is the digital signal-to-distortion ratio in the system best at maximum input level or minimum input level?

Q10-9. What minimum bandwidth would be required for a digitized video signal with a transmission rate of 100 megabits/second, assuming the s/n ratio is 40 dB?

Q10-10. Upon what principle does a compander operate?

Q10-11. Are "information rate" and "transmission rate" the same thing?

Q10-12. What is "headroom"?

Q10-13. Are a RAM and a shift register the same thing?

Q10-14. What is the purpose of PALE?

Q10-15. What must be the minimum storage capacity of the memory in an automatic video synchronizer?

CHAPTER 11

Troubleshooting Digital Circuitry

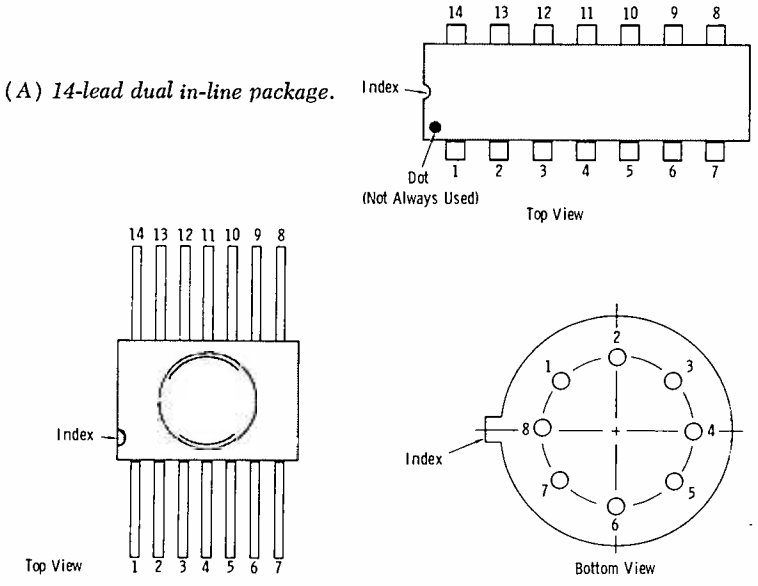
As was stated in the Preface to this book, the text is not a rigorous treatment of the entire field of digital technology. The aim is primarily to effect a "marriage" of the familiar analog field to the less familiar digital world. One of the prime objectives of this book is to help the technician to gain a better understanding of the instruction books that accompany the latest digital equipment and systems.

For this reason, a rather comprehensive testing and servicing procedure will be presented in this chapter. This coverage must necessarily be very general in nature; there is never a good substitute for the specific equipment instruction book. This chapter is based on the assumption that the reader has grasped the fundamentals of digital circuitry covered in the previous chapters. The purpose of this entire text has been to prepare the student to feel at home with troubleshooting procedures that may be covered in equipment manuals.

Just as with vacuum-tube circuitry, when a transistor or IC is suspected of being faulty, direct replacement is the best "check." Plugging transistors and ICs into sockets is an easy check, *providing* care is taken to install the replacement properly. Unlike vacuum tubes, transistors or ICs (chips) can be put into a socket backward. The same problem exists, of course, with transistors or ICs that are soldered into the circuit board. Due to the importance of proper handling of soldered-in ICs, we will start with the final step in troubleshooting: the replacement of the component that appears to be defective.

11-1. IC REPLACEMENT

The three basic types of IC packages, along with their respective terminal arrangements, are shown in Fig. 11-1. Figs. 11-1A and 11-1B



(*A*) *14-lead dual in-line package.*

(*B*) *14-lead ceramic flat-pack.*

(*C*) *TO-5 style 8-terminal package.*

Fig. 11-1. Basic integrated-circuit packages and terminal arrangements.

are top views. Note that pin numbers increase counterclockwise around the package as seen from the top. There are also 16-lead packages that have eight leads per side; pin 16 is opposite pin 1, and pin 9 is opposite pin 8. A package may have 24 or more pins, but the numbering sequence is always the same.

Fig. 11-1C shows the TO-5 type of IC package, for which it is conventional to show the bottom view. Note that the highest lead number is adjacent to the index tab. The numbers then progress clockwise as viewed from the bottom. This type of package may have as many as 12 terminals.

The maintenance technician must be familiar with the proper techniques for replacing ICs and other components on printed-circuit boards. This must be done carefully and skillfully to prevent damage. There are special tools for desoldering all IC pins at the same time (Fig. 11-2). These heated elements are placed on the pins of the IC on the wiring side of the printed board, with a special extractor clamped to the IC body to exert a "pull-away" pressure as the solder is melted. Such tools must be used with extreme caution, since IC leads are sometimes folded against the board and must be bent up before extraction can be done without damage to the board.

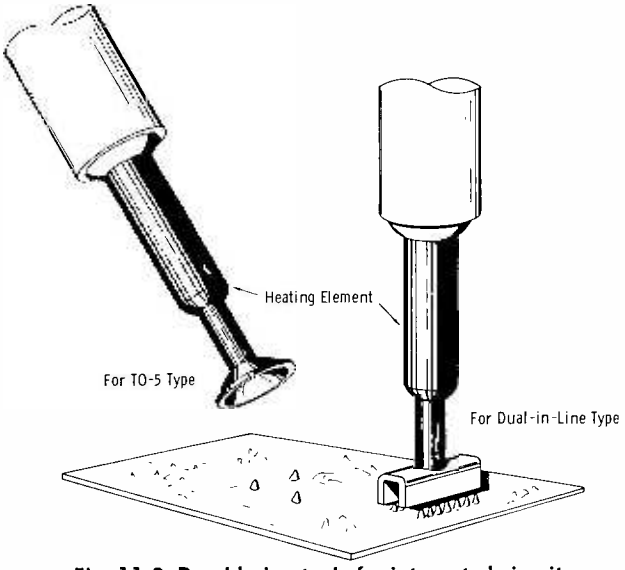


Fig. 11-2. Desoldering tools for integrated circuits.

The preferred method is shown in Fig. 11-3. The procedure is as follows:

1. With the soldering-iron tip applied, squeeze the desoldering bulb and hold its tip at an angle of approximately 45° against the solder fillet of the lead of the component to be removed.
2. As the solder melts, release the bulb quickly to draw solder away; continue to hold the soldering-iron tip against the lead. All solder must be removed from each lead. If the lead is bent against the board, pry it up and repeat the procedure until all solder is removed from the lead.

CAUTION: When the soldering bulb is released to draw in air (and solder), do not remove the soldering iron from the lead. If the iron is removed during this operation, air will cool the joint enough to prevent clean removal of the solder.

3. Repeat the procedure for each of the remaining leads. Note the position of the index (dot, notch, or tab) if the component is an IC. Make certain that all leads are free and clean of burrs; then remove the component. NOTE: Some manufacturers hold flat-packs in place with a small drop of special adhesive that remains flexible and will part under a small amount of pressure. The residue of adhesive from the old flat-pack should be sufficient to hold the new one in place before soldering.

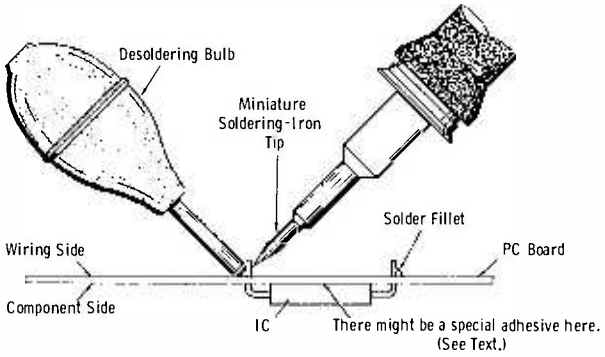


Fig. 11-3. Method for desoldering components.

4. Carefully inspect the holes in the board to ensure freedom from excess solder or burrs that would prevent insertion of the replacement component. Remove burrs if necessary by reaming the holes gently with a sharp instrument such as a pick or soldering aid.
5. Bend the leads of the new component to correspond with those of the one removed. Place the new unit on the component side of the board, making certain that the orienting dot, notch, or tab is in the correct position (see Step 3).
6. Insert the leads through the holes and press the component to the board. Do not trim the leads yet.
7. Solder the leads to the wiring eyelets or pads, using a small-diameter solder to minimize the possibility of solder bridges between leads. Then clip the leads close to the board. Carefully inspect all work after the last lead is soldered to make certain that no solder bridges exist between any leads.

IMPORTANT NOTE: Examine the instruction book for the particular equipment to see if it gives any special instructions with regard to replacement of components.

11-2. BASIC APPROACH TO TROUBLESHOOTING

The first step is to *test* the operation of the equipment. This requires that you know how to operate the unit or system that may be in trouble. Sometimes, defective operation of a system is the result of wrong operating procedures.

Assuming you have the ability to *operate* the equipment, the next step is the obvious one of isolating the defective component. You will need block diagrams of the system, and circuit diagrams that show the ICs and the external hookup of resistors, capacitors, transistors, etc. If you are not familiar with the equipment, study the

labels of all block diagrams involved, and note the important input and output lines that provide fundamental check points.

Some systems have built-in check facilities. If the entire system, including the self-check feature, is faulty, the very first step in isolating the trouble is a power-supply check. Check for proper voltages at all the power-supply outputs. Sometimes, only one voltage of several is wrong. The fault may be in the load or internal to the supply. Remove the boards that receive this voltage one at a time, *with the power off*. If any single board removes the problem, you have, at least, found the defective board. This narrows the problem to troubleshooting the components on that board. If removing the load does not clear the wrong voltage, then the fault is internal to the power supply. Always remember to turn the power off before removing a board, and to turn it off again when reinserting a board.

If the trouble is isolated to the power supply itself, remember that some supply systems have built-in protection such that no voltage is present at any of the outputs in case of certain internal faults. For example, an LED that indicates power on might be driven from a +5-volt output. This +5-volt output may not exist because of a failure in the +15- and/or −15-volt supply where −15 volts is used as a reference for +15 volts, and +15 volts is required for +5 volts.

There are many cases in which a single control function out of many is the only failure. Or, a system fails in only one mode of operation. When you become familiar with the system, you will have a strong hint as to which boards are involved in that particular function. If spare boards are available, replace each board involved, one at a time, until proper operation is restored. Obviously, spare boards pay off in quickly restoring service. If no spares are available, much more time will be required to place the system back in operation. It will be necessary to place each suspected board on an extender (normally provided with the equipment) and check for proper input and output pulses on that board. Review Fig. 10-31 for a typical board arrangement illustrating the plug-in feature.

In tracing a fault, start at the point where the error exists. There may be a light that does not light when it should, or a single control function may fail when the operating procedure says it should function. Start at this point and work back through the circuits involved in fairly large jumps toward the normal origin of the missing or erroneous signal. Eventually, you will find the point at which the signal is blocked or becomes so distorted as to cause a malfunction.

The component where the signal is blocked can be faulty, or another signal on another pin may be missing, preventing the signal from going through. Remember that this might be a dc voltage as well as another pulse. Also, there may be signals present that should

not be, such as preset or clear, or an inhibit signal. This requires further back-checking to find the defect causing the unwanted signal.

Sometimes, an IC will have the proper input signals and an improper output function, yet replacement of the IC does not clear the problem. This is usually due to a shorted or otherwise defective output line.

In troubleshooting digital circuitry, there are two basic test instruments that should be available:

1. The logic test probe
2. The oscilloscope

Logic probes are small, convenient to use, and very inexpensive for normal field use. The oscilloscope is sometimes used for general digital troubleshooting, but it is generally cumbersome in relation to the logic probe. It is, however, mandatory where timing relationships or waveforms must be observed. In MOS circuitry, for example, certain distortions of the clock waveform will prevent proper functioning, and these are best observed on the oscilloscope.

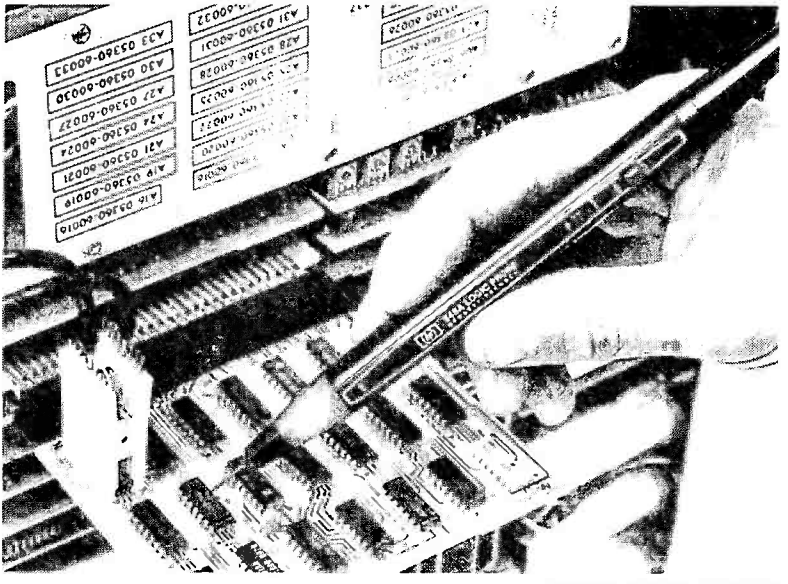
This section has prepared you for a basic approach to digital troubleshooting. Detailed procedures will be covered in Section 11-5, after you become acquainted with the logic probe and oscilloscope as used in troubleshooting digital equipment.

11-3. THE LOGIC PROBE

A logic probe is a small, hand-held pencil-type device that includes a tiny probe and an indicator lamp. It is normally powered from the IC under test with a small IC clip. The latest probes that can handle all logic families and high- or low-level logic designs usually incorporate a switch for selection of the proper family application.

Fig. 11-4 illustrates the Hewlett-Packard Type 545A logic probe in use. This probe has switch-selectable operation. With the TTL-CMOS switch set on TTL, it operates from 4.5- to 15-volt dc power supplies while providing TTL operation. In the CMOS (complementary-metal-oxide semiconductor) position, logic threshold levels are variable; less than 30% of the supply voltage equals "low," and greater than 70% of the supply voltage equals "high." This means that the probe operates for most positive-logic families: TTL, DTL, RTL, CMOS, and MOS.

The probe has an independent, built-in pulse memory and display to catch hard-to-see intermittent pulses. The operator places the probe tip on a circuit point and resets the memory; when a logic change occurs, the memory circuit retains and displays it



Courtesy Hewlett-Packard Co.

Fig. 11-4. Logic probe in use.

until reset. Use of the pulse memory has no effect on normal probe operation; the tip indicator always actively shows the present state of the circuit being probed. Also, interpretation of the simple lamp display is nonambiguous because it can be seen from virtually any angle. Stretching is provided so that short, fast pulses are slowed down for the operator to see. Pulse trains present a similar viewing problem, so the probe blinks at a constant rate for high-repetition-rate signals, allowing the operator to detect the presence of pulse activity.

Fig. 11-5 shows the operating characteristics of the indicator light. A bright glow indicates a one, a dim glow indicates a bad level for the selected logic family, and no light indicates a zero. Short, rapid pulses result in a 10-Hz blinking display.

The pulse memory indicates the first entry into a valid logic level. It also indicates return to the initial valid level from a bad level for a pulse greater than one microsecond wide.

To operate the probe, connect it to the power supply of the circuit under test with grabber-connectors or an IC test clip, and begin to probe around the circuit. Because of its high input impedance and fast speed the probe can be used to troubleshoot CMOS and Schottky devices to find quickly whether a node or gate is stuck, open, or pulsing.

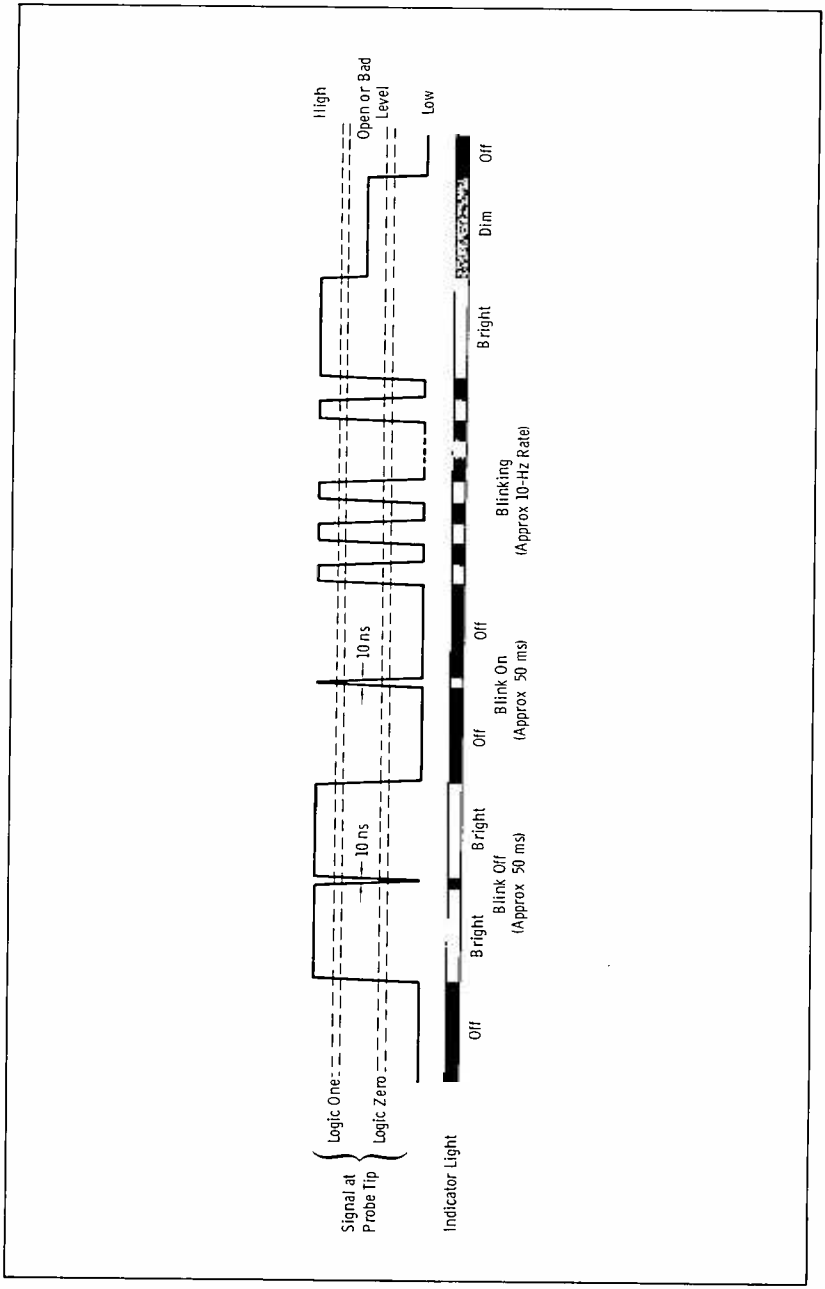


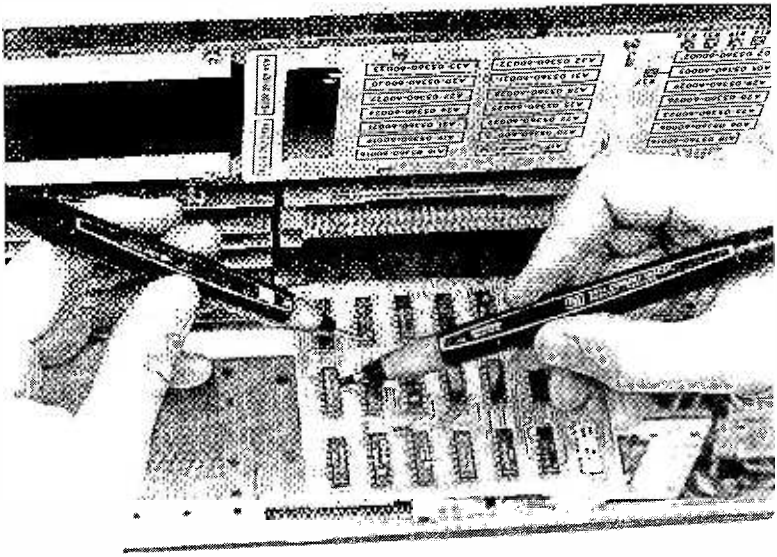
Fig. 11-5. Operation of Type 545A indicator on pulse train.

NOTE: There are many types of logic probes, all operating according to their individual performance specifications. For example, the Model LT-2 (E & L Instruments) has a red LED on at greater than 2.4 volts and a green LED on at less than 0.7 volt. Both LEDs are off for an open circuit (bad level). You will need to become familiar with your particular logic probe as described in the manufacturer's instructions.

The Digital Current Tracer

Prior to the introduction of the logic current tracer, logic-state indicators were limited to displaying voltage information. A node was high, low, open, or pulsing. When a node is stuck, however, it may be trying to change state but not be able to cross threshold levels. Use of the current tracer adds the final bit of information necessary to pinpoint such logic faults on bad nodes. For example, on a bad node, the current tracer can verify that the driver is functioning and also show where the problem is by tracing current to the source or sink that is causing the node to be stuck.

The Hewlett-Packard Type 547A current tracer (on the right) is shown in use in Fig. 11-6. In this photo, the HP 546A logic pulser is also being employed. (The logic pulser is described later.) Constructed as a hand-held probe, the tracer is designed to troubleshoot circuits carrying fast-rise-time current pulses. The tracer senses the magnetic field generated by these signals and displays transitions,



Courtesy Hewlett-Packard Co.

Fig. 11-6. Digital current tracer in use.

single pulses, and pulse trains by means of a built-in one-light indicator.

The sensitivity needed to respond to low-level magnetic fields is provided by a shielded inductive pickup and a wideband, high-gain amplifier. A sensitivity control built into the probe is used to set the lamp brightness to a reference level. During tracing, changes in lamp intensity show the relative level of ac current activity to assist the user in pinpointing the fault location. The 547A senses logic current pulses as small as 1 milliampere, up to 5 millimeters from the conductor.

The tracer operates on all logic families having current pulses from 1 milliampere to 1 ampere with repetition rates up to 10 MHz. Operation does not require interruption of the conducting path or encirclement by a magnetic device. In situations where no signals are present in the circuit, a logic pulser may be used to stimulate the circuit under test; synchronization signals are not required. Note that the primary application of the current tracer is in locating low-impedance faults.

To use the tracer, align the dot on its tip at a reference point, usually the output of the node driver. Set the three-decade sensitivity control to indicate the presence of ac current activity. Then, trace the circuit to see where current is present. As you probe from point to point or follow traces, the lamp will change intensity, and when you find the fault, the tracer will most likely indicate the same brightness found at the reference point. Whether the fault is an undetected solder bridge, a stuck node, or a faulty component with an open input, the presence or absence of current will lead you to it. Obviously, experience is required in the use of any logic probe for maximum results. It is recommended that probes be used on properly operating equipment to gain first-hand knowledge of normal circuit indications with any given probe.

The Logic Pulser

The Hewlett-Packard Model 546A logic pulser (on the left in Fig. 11-6) electrically stimulates integrated circuits belonging to most positive-voltage logic families, such as TTL, DTL, RTL, MOS, and CMOS. Designed as a hand-held stimulus tool for checking gates, counters, shift registers, and flip-flops, the 546A will drive high nodes low or low nodes high over a wide range of supply voltages. The pulser has six output pattern choices that provide a single pulse, a 100-Hz continuous pulse stream, 100-pulse bursts, a 10-Hz continuous pulse stream, 10-pulse bursts, or a 1-Hz continuous pulse stream.

The output of the pulser is three-state. It pulses both high and low, and its 1-megohm impedance when off does not affect the cir-

cuit under test. Automatic pulse-load sensing varies the output so that the right current and pulse width are delivered automatically. Monitor LEDs in the tip indicate which output mode is being used and when each burst occurs. Multipin stimulus kits are available to drive two or more pins simultaneously.

Operating voltage for TTL families is 5 volts dc (±10%); for CMOS it is 3 to 18 volts dc. Operating current is less than 35 milliamperes. Time-base accuracy is ±10%.

The Logic Clip

The logic clip is a device that clips onto all pins of an IC simultaneously. An indicator LED for each pin is mounted on top, allowing up to 16 signals on a single IC to be observed simultaneously. The primary application of the logic clip is in locating a fault in sequential logic.

Each LED has a single threshold level. If a signal (or dc) on a given IC pin is above this level, the corresponding LED in the clip is turned on. If the signal (or dc) is below this level, the LED is turned off. Because of internal power-seeking circuitry, the logic clip cannot be attached improperly. Regardless of how it is clipped onto the IC, it will display the desired signal.

The logic clip differs from the logic probe in two important ways. First, it has a single threshold as opposed to the two thresholds in the logic probe, and it will therefore not indicate a bad level for a given logic family. It will respond to a bad-level signal in the same way that a TTL or DTL gate will: as a logic high state. Second, the logic clip does not have a pulse-stretching circuit. In order to view high-frequency or single-shot narrow pulses, the logic probe should be used.

The ability to view signal activity on several pins simultaneously is a tremendous time-saver. Consider the problem of testing a decade counter. It is necessary to view at least one input and four outputs simultaneously to determine if this device is operating properly. With the logic clip, this can be done conveniently.

11-4. THE OSCILLOSCOPE

The oscilloscope is the star performer for analyzing and troubleshooting analog circuitry. However, its use in troubleshooting digital circuitry is severely limited as compared with logic probes. The "trick" is knowing when to use the scope for tracking down digital problems.

The scope displays a waveshape that is a plot of absolute amplitude versus time. In the digital world, absolute voltage levels are not important. A digital signal exists in one of two states, each de-

termined by a threshold voltage. It is the *relative* value of a signal voltage with respect to these thresholds that determines the state of the digital signal, and this digital state, not absolute levels, determines the operation of the IC. A typical TTL circuit with a 5-volt supply has a high state if the signal is greater than 2.4 volts. It is not important whether the level is 2.65 volts or 2.85 volts. Similarly, for a low state, the voltage must be below 0.4 volt. Again, it is unimportant what the absolute level is as long as it is below this threshold.

It is therefore obvious that a simple on-off (or blinking) indicator light is far more efficient in fast troubleshooting of digital circuitry than an oscilloscope is. Even timing characteristics within a digital logic family, such as TTL, are well defined for each component. Each gate in the TTL logic family has a characteristic propagation delay time, rise time, and fall time. When a digital IC fails, it generally fails catastrophically. This means that timing parameters rarely degrade or become marginal. Thus, any advantage in using the scope for digital servicing is minimal, due to the natural digital-world environment.

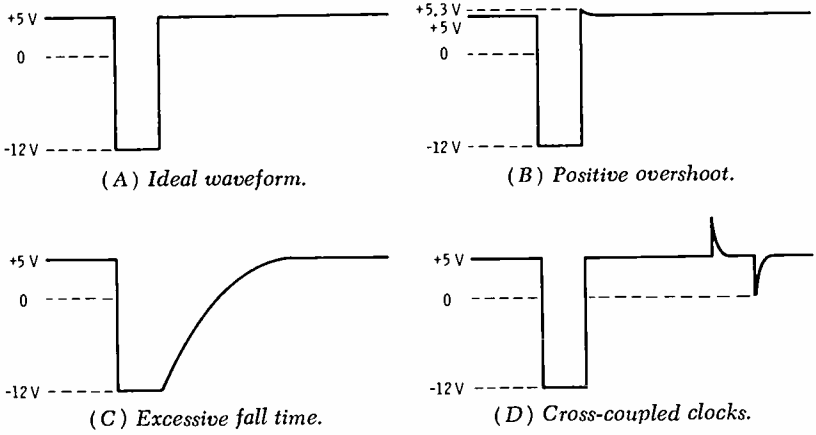
There are exceptions, however, to almost any rule, and MOS devices in particular require clock signals of the proper amplitude, shape, and timing. Although these problems should have been corrected in initial design, occasionally an improper or partially defective external resistor or capacitor will cause short overshoots or undershoots on a clock waveform. If you ever find an MOS IC that does not function even though all checks show that it should (assuming you have determined that the output line is not shorted), check the clock waveform.

Clock Waveforms

In many MOS devices, the clock signal must swing from −12 volts to +5 volts. The clock generator must provide a clean waveform that has reasonable rise and fall times (usually under 40 nanoseconds) and does not have positive overshoot. The scope is mandatory in checking such characteristics.

With a choice of the proper voltage scale, the displays of Fig. 11-7 can be observed by using the dc scope position. The following characteristics apply (the letters that identify the waveforms refer to the parts of Fig. 11-7):

A. This is the ideal clock waveform.
B. Indicates an overshoot occurring on the positive-going clock transition. This has the effect of forward-biasing the substrate diode and must be avoided to prevent erratic behavior in the driven device.



(A) *Ideal waveform.* (B) *Positive overshoot.*

(C) *Excessive fall time.* (D) *Cross-coupled clocks.*

Fig. 11-7. Clock waveforms.

C. The clock requires excessive time to return to 5 volts (zero reference). Data may toggle through a shift register but may not be stored.
D. Indicates cross-coupling between two clock drivers. This is usually caused by lack of clamping or nonactive (high-impedance) switching in the positive direction.

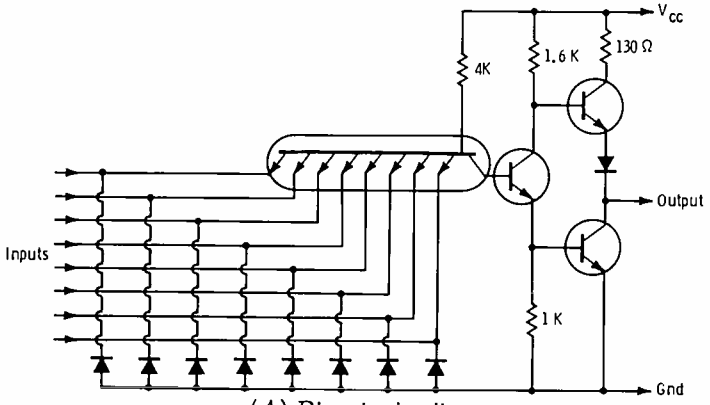
Timing Information

Most clock generators for multiple-phase waveforms employ fixed-logic circuitry. In this case, either the two or more output phases are correct or the unit does not function at all. However, you may encounter a multiple-phase clock that employs one-shots with adjustable pulse widths. The dual-trace or four-trace scope (if the clock is four-phase) provides the most convenient check. The scope is operated with external trigger from phase 1 so that timing relationships can be observed. Pulse-width adjustments are then made according to the particular system requirements.

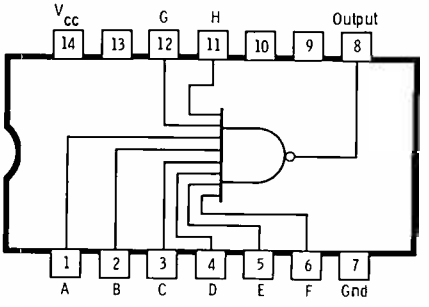
11-5. TROUBLESHOOTING PROCEDURES

The basic approach to digital troubleshooting techniques was covered in Section 11-2. Now that you are familiar with the logic probe and the digital use of the oscilloscope, we can go ahead with more detailed procedures.

Fig. 11-8A illustrates a typical discrete circuit of an eight-input NAND gate. A multiple-emitter transistor (MET) is used for the input signals. Each diode, resistor, and transistor could be tested by using a signal generator, voltmeter, ohmmeter, or oscilloscope.



(*A*) *Discrete circuit.*



(*B*) *IC schematic.*

Fig. 11-8. TTL 8-input NAND gate.

But when this circuit is built in IC form (Fig. 11-8B), the individual components are no longer accessible. The traditional troubleshooting tools are now of only limited value.

All you have available are the eight inputs and single output, and your knowledge that the output will be low *only* when all inputs are simultaneously high. What you are working with is the truth-table function of the NAND gate.

When you have a complex system consisting of logic gates, flip-flops, shift registers, memory banks, etc., *always* start with the nearest gate that is involved with the particular output error. A gate has a simple function that is clearly evident from the truth table for the particular gate. Checking the gate function nearest to the defective output will tell you whether to proceed backward or forward in the path involved.

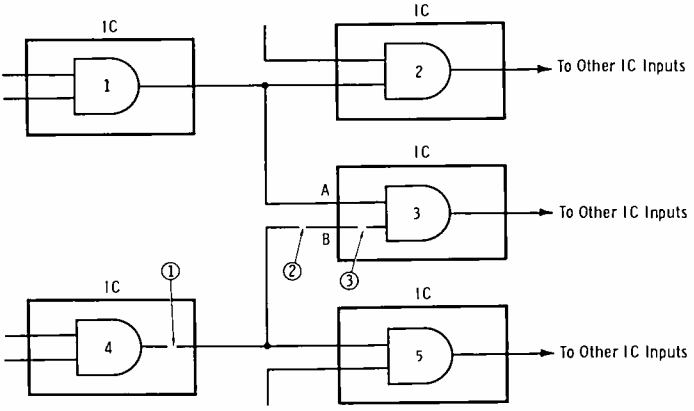
Effect of Opens

As was stated previously (Section 3-5), the internal connections in a gate will allow a "floating" (open) input to rise to around 1.2

to 1.8 volts. In any TTL or DTL IC, this "bad level" (neither the low threshold level of 0.4 volt nor the high threshold level of around 2 volts) will generally be interpreted as a high logic level. Thus, an open (either internal or external) to an IC gate input will normally result in an operation that indicates the particular input is at a high level.

Refer to Fig. 11-9 and assume that only one open can occur at a given time. Point 1 represents an open output bond internal to gate 4. All inputs driven by that output are left to float. In Fig. 11-9, gates 3 and 5 and all ICs driven by them would be affected. An open at point 2 is external to gate 3. This would be evident by a signal at the output of gate 4 and the input of gate 5, but not at the B input of gate 3. Gate 3 and all ICs following it would be affected. An open of the internal bond of gate 3 (point 3) would have the same effect as the open at point 2, except that the B input of gate 3 would show normal.

Now let us see what a logic probe such as the HP 545A (described in Section 11-3) would tell you. Assume you have the open at point 1, which is an open internal bond at the output of gate 4. The indicator probe at this output terminal indicates the gate is stuck either high or low, depending on the internal structure of the IC and where the open is located. The important characteristic is that the output is not changing with the input signals (which you determine are normal by means of the logic probe). The lamp in the probe will remain bright if the output is stuck in the high level, dim if it is stuck in the bad level, and extinguished if it is stuck in the lowest (zero) level.



Note: Assume only 1 open to occur at a time.

Fig. 11-9. Effect of opens.

Similarly, if the open is at point 1, the indicator at the associated inputs of gates 3 and 5 will have the same unchanging characteristic as at the output of gate 4. Usually, the output of gate 4 will show a dim glow on the logic-probe indicator; this is a "feedback" of the bad-level condition at the inputs of gates 3 and 5.

Have you concluded now that the IC containing gate 4 is faulty and should be replaced? No! This gate could be stuck high due to an external short of the output line to V_{CC}. It could be stuck low due to an external short of the output line to ground. Therefore, a check must be made on the output line before the IC is replaced, particularly if the IC is a type that is soldered in rather than plugged in. Remember also that shorts of this nature can occur on the inputs internal to the driven ICs. This can be determined quickly if the ICs are plug-in types by removing the driven ICs from their sockets and checking the gate-4 output to see if the stuck condition is cleared. In the case of soldered-in ICs, more checks for shorts should be made as described later.

Now assume there is an open at point 2 of Fig. 11-9. The output of gate 4 and input of gate 5 will show normal operation. At the B input of gate 3, the probe will normally indicate a bad level, and the general location of the open will become obvious. In this case, the effect will be as if the B input is at high level, and therefore the output of gate 3 will depend on the A input.

Note carefully that an open of the output bond inside the IC (point 1) will cause the output to be stuck. This can also be caused by a short on the output line, either by a solder bridge between printed-circuit paths or by an internal short in the input circuit of a following IC. An open in the printed-circuit path external to an IC (point 2) will cause a bad-level indication at the corresponding input pin of the IC, and the gate operation will be as if that input is at a static high level; thus, the output state will depend entirely on the other input(s). An open of the input bond inside the IC (point 3) will not be indicated at the corresponding input pin, but (again) the output will depend entirely on the other input(s).

In checking for opens in printed-circuit paths, a magnifying glass is often required to detect the tiny hairline cracks that usually occur. If the IC is on a removable board, flexing the board while observing it under the magnifying glass will reveal the break.

Effect of Shorts

A short between any node (accessible operating point) and V_{CC} or ground external to the IC is indistinguishable from a short internal to the IC. Either will cause the signal lines connected to the node to be stuck either high (for shorts to V_{CC}) or low (for shorts to ground). When this type of failure is encountered, a rigorous physi-

cal examination of the printed wiring paths is required to isolate the cause. Solder bridges are commonly encountered. Examine closely all soldered points on the circuit board that are in close proximity to another conductor. Solder "tails" can usually be found by close inspection. A magnifying glass may be necessary to detect hairline shorts. If you find one, run the sharp point of a scriber gently between the runs to remove the short.

Fig. 11-10 shows the three most commonly encountered kinds of shorts, as well as the typical totem-pole output circuitry of DTL and TTL logic ICs. A short at point 1 will hold line A at high level, and a short at point 2 will hold line B at low level. These shorts are readily detectable with a logic probe.

A short between input pins (point 3) is not as straightforward to analyze as a short to V_{CC} or ground. This is why we have included the typical output circuitry of logic ICs for the purpose of analysis. Whenever the outputs of IC1 and IC2 go high simultaneously or go low simultaneously, the shorted pins will respond properly. However, if one output tends to go high while the other output tries to go low, the circuit operation will fail. The output

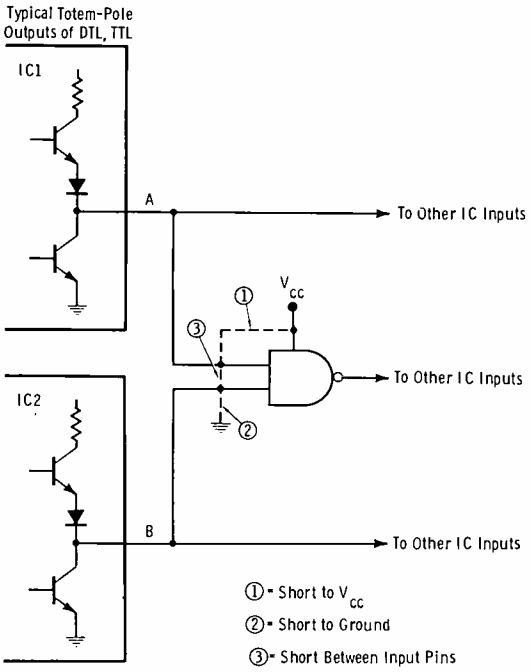


Fig. 11-10. Effect of shorts.

attempting to go high will supply current through the upper saturated transistor of its totem-pole output stage, while the output attempting to go low will sink this current through the saturated lower transistor of its totem-pole output stage. The saturated transistor to ground pulls the shorted pins to a low state. Note that the output of the NAND gate shown in Fig. 11-10 would be stuck in the high state (it is low only when both inputs are high). If the gate were an AND circuit, the output would be stuck low. Also note that none of the following ICs, whether driven from line A, line B, or the NAND gate, would function properly.

In tracing high-impedance paths (opens), the straight logic probe is most efficient. It is also highly efficient in locating the fault caused by a short. However, for *tracing* the low-impedance path caused by a short, the current tracer is most efficient.

In lieu of a current tracer, a sensitive ohmmeter with milliohm resolution can be used. Turn off the power, and remove the circuit board if possible. Using the lowest scale on the ohmmeter, put one probe on the input or output pin and the other on ground or V_{CC}, depending on the nature of the short. Hold the probe on the ground or V_{CC} point very steady, and move the other probe from the IC pin along the printed-circuit run. You will notice a decrease in resistance as you approach the locality of the short and an increase in resistance as you move away from the short. Move to the point of minimum resistance. Now repeat the procedure using the other probe to find the minimum resistance. The short will be very close to this point.

An IC can be stuck in the high or low mode because of failure of the internal steering circuitry. This will permanently turn on either the upper transistor of the output totem pole (locking the output high) or the lower transistor (locking the output low). This is a fault internal to the IC and not in external printed wiring or input circuits of driven devices. With plug-in ICs, the best way to check this is to remove the IC from the socket and bend the output pin upward directly from the point of the original bend. Reinsert the IC in the socket and check the output. If the output is still stuck, a defective IC is indicated. In the case of ICs soldered into the printed-circuit board, it is better to double-check for shorts before replacing the IC.

> CAUTION: For plug-in ICs, always check directly at the IC pin rather than a contact on the socket. Then check at the socket terminal on the wiring side. This will reveal any possible socket defect—which might cost many hours of unnecessary servicing time if not detected.

In the TTL family of ICs, there is a subfamily called *open collector gates.* (Review Section 3-11.) Collectors can be "wire-ORed" by

the arrangement of Fig. 11-11. With this configuration, the output of any one of the gates can constrain the outputs of the other ICs to be in a state other than that defined by the truth table for the given input states. For example, a defect in gate 1 can cause the outputs of gates 2 and 3 to operate improperly.

If the ICs are the plug-in type, bend the output pins upward (to remove the load), and reinsert the ICs so that they may be checked individually. The defective gate will now be apparent and can be replaced. When the ICs are soldered into the circuit board, you have only one course of action. Try to locate any short by using either the current tracer or a sensitive milliohmmeter as described above. If this does not reveal the fault, replace each IC one at a time until proper operation is restored. All gates of a wired-OR configuration may, of course, be contained in one IC, saving you considerable time.

Summary

A summary of Sections 11-2 and 11-5 follows.

1. Become so familiar with basic AND gates, OR gates, and flip-flops that you can construct a truth table for the specific application (Chapter 3).
2. Be certain you know the proper way to set up and operate the equipment being serviced.
3. Study the block diagrams of the system to get a comprehensive picture of signal flow. Note the labels of input and output lines on the schematic of each module.
4. Start with the end result of a faulty function, and work back from this point.
5. When working back from the faulty output, jump back (if necessary) to the last logic gate involved. Actions of logic gates are relatively easy to interpret.
6. If spare circuit boards are available, substitute a board suspected to be faulty.
7. If a particular IC is suspected of being faulty, check its temperature with your finger *if there is no shock hazard.* Very often, an internal fault will cause a drastic temperature rise.

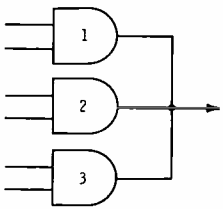


Fig. 11-11. Open-collector AND gates in wired-OR configuration.

8. For a plug-in IC, substitute a new one, being very careful to insert the new IC in the socket properly. Use identical replacement.
9. When a line is stuck low, check for a short to ground or between pins. When a line is stuck high, look for a short to V_{CC}. Remember also that a stuck line can be caused by a defective IC driving the line or a defective input on a driven IC.
10. Probe the pins on the device, not the terminals on the socket. To detect a faulty socket, compare the readings on the pins with the readings on corresponding socket terminals on the wiring side.
11. For temperature-sensitive intermittent problems, use a small hair dryer to apply heat to a suspected component. Or use a can of spray coolant on a hot board. The blast of coolant should be brief and confined to the suspected component.
12. When all else fails, never hesitate to contact the field service department of the equipment manufacturer. Quite often, a brief telephone conversation will provide the clue for the solution of a particular problem.

11-6. MEASUREMENTS ON DIGITIZED VIDEO SYSTEMS

Due to the fundamental nature of digitized video signals, special testing techniques are required to make meaningful performance measurements.

You should recall from previous coverage of digitized audio and video signals that quantizing error and quantizing noise are generally taken to be one and the same. In digitized video systems such as time-base correctors and automatic frame synchronizers, this statement must be modified to account for the fact that tv signals are synchronous rather than random, and do have noise. Much of the following material is based on research by Michael O. Felix, chief engineer of the Audio/Video Systems Division of the Ampex Corporation, and it is reproduced here with the permission of Ampex.

Some traditional video measurement techniques are not well suited to digital video systems. For example, the standard procedure for measuring differential phase (dp) and differential gain (dg) is with a test-signal subcarrier level of 20 IEEE units. In a digital system, this amplitude is defined by only a small number of digitization levels, and the quantizing errors encountered are significant compared with the test signal. These could be interpreted as large system errors, when in fact the errors are insignificant when compared with normal signal amplitudes, and not discernible on a picture tube. This test method as originally contrived was really a technique for quantifying vacuum-tube nonlinearities.

Put another way, digital systems are at their best when handling full-amplitude signals. The subjective effects of dp/dg become most apparent when the chroma level is near saturation, and at this point the effects of quantizing error are at their lowest. At low levels (below 40 IEEE units), measured dp/dg becomes poorer due to quantizing, but at that level the effects are no longer visible. Thus, far more accurate test results are obtained at chroma levels of 40 IEEE units in a digital system.

Signal-to-noise ratio measurement is likewise a misunderstood subject in digital systems. Quantizing error is often confused with quantizing noise. Quantizing error represents the smallest differential between one increment of measurement and the next in an analog-to-digital converter. In a well-designed system, it is plus or minus one-half the least significant bit (LSB), and in an 8-bit digital system it amounts to $\frac{1}{512}$th of the full signal swing. In a perfectly quiet and random signal, quantizing error does equal quantizing noise, but tv signals are not random and do have noise. Further, much of the noise that is added by the digital conversion process lies outside the video passband. Thus, digital noise based on a calculation of quantizing error is not accurate, and in a high-quality digital system, the apparent noise contribution is 8 to 10 dB less than the calculated value based on a random and quiet signal.

The conventional method of shorting the input and measuring the noise at the output to determine s/n ratio is likewise inaccurate. A digital system will behave quite differently with a real signal with noise on it. In a digital video system, the most effective method of determining signal-to-noise ratio is by measurement of noise contribution to a video signal, which in a carefully designed 8-bit system can approach −60 dB when the usual test signals are used.

The following information is from a paper by Michael O Felix, "Differential Phase and Gain Measurements in Digitized Video Signals."[1]

Introduction

The common test signal for measuring dp and dg in a vtr is a 20-IEEE-unit subcarrier superimposed either on a 100-IEEE-unit ramp (Fig. 11-12) or on a stairstep signal. Typical specifications in broadcast vtr's run from 1 to 4% dg and 1 to 4° dp.

Such a test signal does not fully load the circuits, which, with saturated colors, must handle subcarrier swings of 126 IEEE units, more than six times greater. Under these conditions the distortions

1. Reprinted by permission from the February 1976 *SMPTE Journal*, © 1976 by the Society of Motion Picture and Television Engineers, Inc., 862 Scarsdale Ave., Scarsdale, NY 10583.

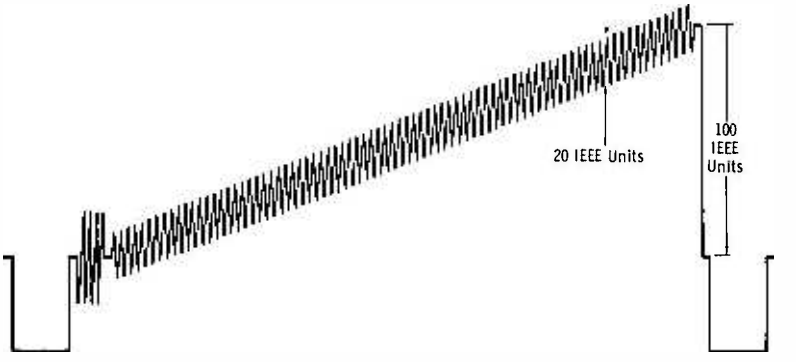


Fig. 11-12. Standard dp/dg test signal for analog devices.

will be greater, but practical experience has shown that there is a consistent relationship between the dg/dp measurements and the visual quality of the saturated picture.

Quantizing errors in digital systems have very different characteristics, for they introduce errors that have a fixed peak value irrespective of the signal quantized. For example, a typical 8-bit system can introduce a ±0.3-IEEE-unit error on a 6-IEEE-unit signal (an error of ±5%), but on a 100-IEEE-unit signal, the error can still not exceed ±0.3 IEEE unit, or ±0.3%.

Therefore, if quantizing errors are made invisible on a low-level signal, they will be invisible on a high-level signal. Contrast the analog case where the above 5% distortion of a 6-IEEE-unit signal would correspond to an intolerably high distortion on a 100-IEEE-unit signal.

Bit Levels in NTSC and PAL Signals

The most widely used digital standard found in digital video circuits is a code based on eight bits per sample and three samples per subcarrier cycle. In an NTSC or PAL signal, the subcarrier can extend from −33 IEEE units to +133 IEEE units (Fig. 11-13). This peak-to-peak swing of 166 IEEE units must then be covered by the 256 levels (255 steps) that eight bits can provide. Thus, the step size corresponding to a change in the LSB (least significant bit) is 166/255 = 0.651 IEEE unit.

Quantizing Errors

Consider an NTSC signal corresponding to a uniform color picture. The video waveform (Fig. 11-14) is a 3.58-MHz sine wave superimposed on a fixed voltage that corresponds to the brightness.

Each cycle of the waveform is sampled three times (Fig. 11-15), the samples being accurately spaced by 120°. However, the points

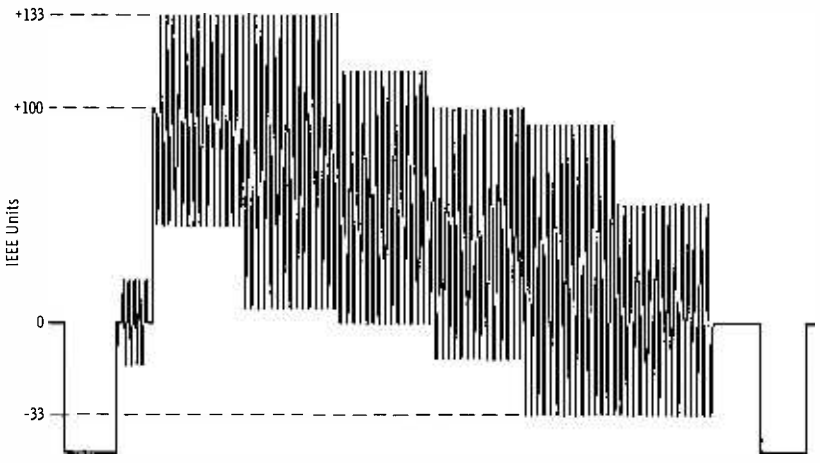


Fig. 11-13. Color bars at 100%.

in the cycle where the samples occur depend on the color that the signal represents.

Each sample is now quantized; that is, from here on it is represented as the nearest level of the 256 available. Fig. 11-16 shows the situation with the spacing between levels exaggerated, and it can be seen that the quantized level may differ from the true one by $\pm\frac{1}{2}$ LSB.

The only information passed to the following system is the sample timing and its quantized value. From this must be reconstructed as close an approximation to the original sine-wave signal as possible. This can be done by generating short pulses having heights proportional to the quantized values and feeding this pulse train through a low-pass filter (Fig. 11-17).

The resulting sine wave can differ from the applied waveform in three ways. Normally, all three errors are present, but special cases can be considered that isolate them. These cases are illus-

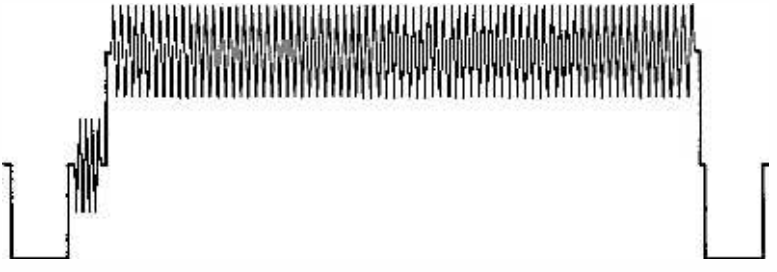


Fig. 11-14. Line waveform for color of fixed hue and saturation.

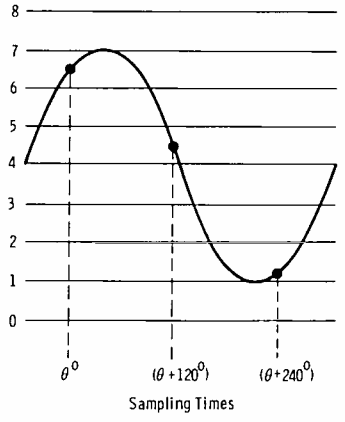


Fig. 11-15. Sampling of one subcarrier cycle.

trated in Fig. 11-18. In each case, the solid waveform is the input value, and the dashed waveform is the quantized value. (The differences between the waveforms have been exaggerated.) For example, if all three quantized levels are equally too high (Fig. 11-18A), the result is a dc shift of the waveform. If the errors are symmetrical about the center of the sine wave, the result is a change in amplitude (Fig. 11-18B). A particular form of asymmetric error can result in a phase shift without change in either amplitude or dc component (Fig. 11-18C). In television terms, Fig. 11-18A shows chrominance-to-luminance cross talk; Fig. 11-18B shows a chrominance gain error; and Fig. 11-18C shows a shift in chrominance phase.

DP/DG Measurements

Assume a standard dp/dg test is applied to an ideal, noise-free digital video system. It can be shown that on a test signal with a peak value of 10 IEEE units (20 IEEE units peak-to-peak), the

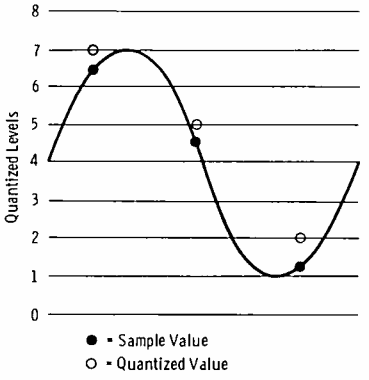


Fig. 11-16. Quantizing of subcarrier cycle.

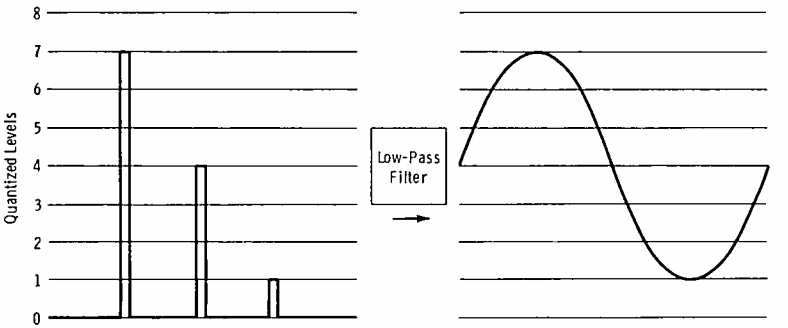
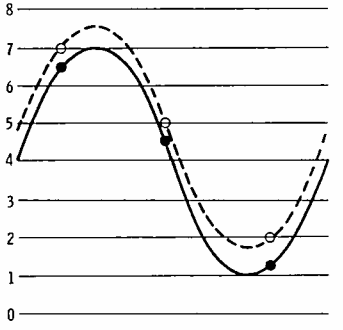


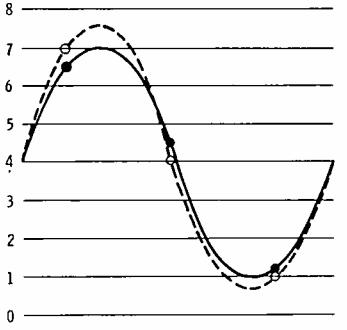
Fig. 11-17. Reconstruction of original waveform.

maximum error can be ±4.3% dg, ±2.5° dp, and ±0.33-IEEE-unit luminance shift.

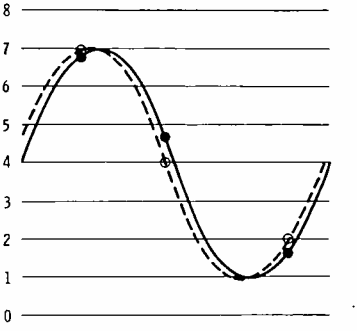
Consider first the dg error. To have the worst-case error, four conditions must be met simultaneously:



(A) Chrominance-luminance cross talk.



(B) Chrominance gain error.



(C) Chrominance phase shift.

Fig. 11-18. Three cases of quantizing error.

A. The relative timing between the test signal and the sampling must be such that the samples occur at 30°, 150°, and 270°.
B. The first sample value must be just above a critical level so that its quantized value is ½ LSB too high.
C. The second sample must also lie just above a critical level.
D. The third sample must lie just below a critical level so that its quantized value is ½ LSB too low.

All four conditions rarely occur simultaneously, and this accounts for the "spiky" nature of dg measurements on digital systems. The average error is much lower, but occasionally these larger errors occur. Note, however, that even ±4.3% of 10 IEEE units is less than one IEEE unit peak-to-peak and that, in contrast to the analog case, this absolute error does not increase on larger swings, and the percentage error decreases proportionately.

A similar situation exists for the dp errors. The conditions for maximum error are different from those for dg (sampling times must be 0°, 120°, and 240°, for example) so that maximum dg and dp cannot occur simultaneously. In fact, when one reaches its maximum, the other is zero. Here again, there is a parallel between dp measured in degrees and dg measured in percent—both decrease proportionately on larger swings.

The effect of these quantizing errors is invisible on a first-generation tape recording under the most critical of test conditions. Even multigeneration effects are negligible in a system with a 50-dB s/n ratio, where the noise degradation hides the quantizing errors. Contrast the results from an analog system with 8.6% dg and 5° dp, which would be poor as a first generation and unacceptable after several generations.

Effects of Noise

The effects of noise can be divided into two parts. If the noise is large enough to carry the signal through several levels, then a quantized version of the noise will be added to the output. A signal-to-noise ratio of 46 dB means the rms noise is ½ IEEE unit. The peak value is therefore 1½ IEEE units, and the peak-to-peak value is 3 IEEE units. With the LSB equivalent to 0.651 IEEE unit, the noise extends through nearly five levels, and a similar noise will be found at both input and output.

There is a second effect more clearly seen at very high signal-to-noise ratios. Quantized systems are characterized by having "critical levels," levels at which the system cannot decide whether to go to the next higher or next lower level. Even with an 80-dB signal-to-noise ratio at the input, if the system is precisely adjusted, there will be a peak-to-peak output noise of 1 LSB—in our example, 0.651

IEEE unit or roughly 60 dB rms. Any slight shift in the input level, however, will result in no output noise whatsoever, since the input noise amplitude is too small to take the system over the threshold level. In practical systems that move through these points, noise "blurs" the effect and makes the critical levels less critical. Note, however, that the ±4.3% dg figure given earlier remains unchanged as the peak error; the system merely cannot be set to produce this error continuously, resulting in a lower average value. An oscilloscope will still show the same peak spikes.

This second effect makes it difficult to define the noise introduced, for example, by a digital time-base corrector (DTBC). One cannot feed a gray-level signal into the DTBC and measure the output noise, since it may read −60 dB at one input level and −75 dB at another only 0.3 IEEE unit higher. A common method, therefore, is to connect the DTBC in series with the vtr and to measure the degradation; one can then calculate the "equivalent s/n ratio" of the DTBC. If, for example, a vtr has a 50-dB s/n ratio, which the DTBC degrades to 49.5 dB, the noise power has been increased in the ratio of 1:1.12. The equivalent DTBC noise is therefore 0.12, or 9.2 dB, better than the vtr, and the DTBC equivalent s/n ratio would then be 59.2 dB. This is not an accurate method, since it depends on small differences between large numbers, but it does appear to be the best approach to providing a meaningful figure.

Alternate Methods of Measuring DP/DG

The use of a signal of 20 IEEE units peak-to-peak does not properly evaluate the performance of the digital circuits within a vtr. Errors in the analog recorder that are important to the final picture quality can be masked by quantizing errors that are finally invisible.

Two steps are proposed to improve this situation, although they will not entirely correct it. The first is to double the subcarrier amplitude to 40 IEEE units peak-to-peak. The second is to make the subcarrier phase random during the ramp or stairstep.

The change to 40 IEEE units halves the dp/dg errors, making them ±1.2° and ±2.1%. The use of a random-phase subcarrier means that the spikes corresponding to the peak values of dp or dg are no longer locked either to line rate or to frame rate. The resulting oscilloscope display is therefore smoother, and it is much easier to see the normal gradual dp/dg changes of the analog vtr circuits through the spiky quantizing effects.

Summary

When the standard method of measuring dp/dg is applied to digital circuits, very misleading results are obtained. A digital sys-

tem capable of high-quality multigeneration signals will show errors of over 8% dg and 5° dp. Not only are these results misleading, but when the digital and analog circuits are in series (for example, a digital time-base corrector in an analog vtr), the digital effects can mask the performance of the analog system; serious errors in the analog system may be almost completely hidden.

Two improvements in measuring techniques are proposed. First, the subcarrier amplitude of the test signal should be doubled, and, second, its phase should be made random. These two changes do not eliminate the problem, but they will make it much easier to take measurements of a combined analog/digital system.

EXERCISES

Q11-1. Why is the occurrence of pulse activity as revealed by a logic probe usually sufficient indication of proper operation?

Q11-2. The three inputs to an error-indicating NAND gate are high, low, and low for correct system operation in a particular mode. How can you check this NAND gate for proper function without simulating an error that would cause all the inputs to be high?

Q11-3. What is the most convenient way to diagnose a problem at the output pin of an IC?

Q11-4. Give three possible ways in which a node could be shorted to V_{CC} or ground.

Q11-5. How will a logic probe indicate an open circuit prior to the input pin of a logic IC?

Q11-6. What type of probe would you use to observe two or more simultaneous actions of a logic IC?

Q11-7. What is meant by a "pulse memory" in a logic probe?

Q11-8. When is the use of the oscilloscope mandatory in checking digital circuitry?

Q11-9. In a digitized audio or video system, does quantizing error change between low- and high-level signals within the operating range of the system?

Q11-10. In testing a digitized video system, what modification of the test signal is recommended for measuring differential gain and phase as opposed to that used for an all-analog system?

APPENDIX A

Glossary

This glossary covers those basic terms employed in the text of this book. Definitions are based on various United States Government publications and the USASI *Glossary of Computer Terms.*

Abort: In a computer, the condition that results in the skipping of the next sequential instruction.

Access Time: The time required for a digital unit to locate data or an instruction word in the memory section and transfer it to the control unit where the required controls are performed.

Accumulator: The register and associated equipment in the arithmetic unit of the computer in which arithmetical and logical operations are performed.

Accuracy: The degree of exactness of an approximation.

Acknowledge: Indication of the status of data on the input/output lines. Abbreviated ACK.

Adder: A device that can combine and indicate the sum of two or more numbers or quantities.

Address: A coded number that specifically designates a register or other internal storage location. Information is referenced by its address. Portions of control are responsible for directing information to or from an addressed location.

Algebra, Boolean: A process of reasoning, or a deductive system of theorems using a symbolic logic and dealing with classes, propositions, or on-off circuit elements. It employs symbols to represent operators such as AND, OR, NOT, etc., to permit mathematical calculation. Named after George Boole, an English mathematician (1815–1864).

Algorithm: A fixed step-by-step procedure for solving a problem.

Alphanumeric: A contraction of alphabetic-numeric.

AND: Same as *Operator, AND.*

AND Circuit: Same as *Gate, AND.*

AND Gate: (*See Gate, AND.*)

AND Operator: (*See Operator, AND.*)

Arithmetic: A section within the computer where reasonable processes such as addition, subtraction, multiplication, and division are performed, and operands and results are stored temporarily.

Arithmetic, Fixed-Point: A method of calculation in which operations take place in invariant manner, and in which the computer does not consider the location of the radix point. This is illustrated by desk calculators or slide rules, in which the operator must keep track of the decimal point. Similarly, in many automatic computers the location of the radix point is the programmer's responsibility. Contrasted with *Arithmetic, Floating-Point.*

Arithmetic, Floating-Point: A method of calculation that automatically accounts for the location of the radix point. This is usually accomplished by handling the number as a signed mantissa times the radix raised to an integral exponent. For example, the decimal number $+88.3$ might be written as $+0.883 \times 10^2$, and the binary number -0.0011 as -0.11×2^{-2}. Synonymous with "floating-decimal arithmetic"; contrasted with *Arithmetic, Fixed Point.*

ASCII: American Standard Code for Information Interchange.

Asynchronous: Pertaining to a lack of time coincidence in a set of repeated events. This term is applied to a computer to indicate that the execution of one operation is dependent on a signal that the previous operation has been completed.

Base: Same as *Radix.*

Binary: A characteristic, property, or condition in which there are but two possible alternatives, e.g., the binary number system using 2 as its base and using only the digits zero (0) and one (1).

Binary-Coded Decimal: Notation in which each element of the decimal system is represented by a fixed number of binary positions. Abbreviated bcd.

Binary Number System: A number system with two symbols (0 and 1) that has 2 as its base just as the decimal system uses ten symbols (0, 1, . . . , 9) and a base of 10. (*See also Positional Notation and Radix.*)

Bistable: The capability of assuming either of two stable states, hence, of storing one bit of information.

Bit: A binary digit, 0 or 1, represented by the condition (set or clear) of a stage.

Bit Rate: (*See Rate, Bit.*)

Boolean Algebra: (*See Algebra, Boolean.*)

Byte: (1) A generic term that indicates a measurable portion of consecutive binary digits, e.g., an 8-bit or 6-bit byte. (2) A group of binary digits usually operated upon as a unit.

Capacity: The upper and lower limits of the numbers that may be processed in a computer register.

Capacity, Memory: The number of elementary pieces of data that can be contained in a memory or storage device. Frequently defined in terms of characters in a particular code or words of a fixed size that can be so contained.

Card Code: A combination of punched holes that represent alphanumeric and special characters in a punched card.

Card Punch: A device used to record information by punching holes (generally rectangular) in cards.

Card Reader: A device that senses and translates the holes in punched cards into binary form.

Character: One symbol of a set of elementary symbols such as those corresponding to the keys on a typewriter. The symbols usually include the decimal digits 0 through 9, the letters A through Z, punctuation marks, operation symbols, and any other single symbols that a computer may read, store, or write.

Clear: To restore a storage or memory device to the zero state.

Clock: (1) A master timing device used to provide the basic sequencing pulses for the operation of a synchronous computer. (2) A register that automatically records the progress of real time (or some approximation to it) and records the number of operations performed, and whose contents are available to a computer program.

Clock Rate: (*See Rate, Clock.*)

Code: (1) A system of symbols for meaningful communication. (2) A system of symbols for representing data or instructions in a computer or a tabulating machine. (3) To translate the program for the solution of a problem on a given computer into a sequence of machine language or pseudoinstructions and addresses acceptable to that computer. (4) A machine-language program.

Coded Program: A procedure for solving a problem by means of a digital computer. The program may vary in detail from a mere outline of the procedure to an explicit list of instructions coded in machine language.

Coincidence Gate: A circuit with the ability to produce an output that is dependent on a specified type of or the coincident nature of the input. For example, an AND gate has an output pulse when there are pulses in time coincidence at all inputs; an OR gate has an output when any one or any combination of input pulses occurs in time coincidence. A gate may contain a number of inhibits; there is no output under any condition of input if there is time coincidence of an inhibit or except signal.

Command: One of a set of signals or groups of signals resulting from an instruction. Commands initiate the individual steps of the instruction.

Comparator: A device for comparing two different transcriptions of the same information to verify the accuracy of transcription, storage, arithmetic operation, or other processes; a signal is given dependent on some relation between two items (i.e., one item is larger than, smaller than, or equal to the other).

Complement: A quantity, expressed to the base n, that is derived from a given quantity by a particular rule. It is frequently used to represent the negative of the given quantity. A complement on n may be obtained by subtracting each digit of the given quantity from n − 1, adding unity to the least significant digit, and performing all resultant carries. For example, the twos complement of binary 11010 is 00110; the tens complement of decimal 456 is 544. A complement on n − 1 may be obtained by subtracting each digit of the given quantity from n − 1. For example, the ones complement of binary 11010 is 00101; the nines complement of decimal 456 is 543.

Computer: A device capable of accepting information, applying prescribed processes to the information, and supplying the results of these processes. A computer usually consists of input and output devices; storage, arithmetic, and logical units; and a control unit.

Computer, Asynchronous: A computer in which the performance of each operation starts as a result of a signal either that the previous operation has been completed or that the parts of the computer required for the next operation are available. Contrasted with *Computer, Synchronous.*

Computer, Digital: A computer that processes information represented by combinations of discrete or discontinuous data (as compared with an analog computer for continuous data). It is a stored-program device capable of performing sequences of internally stored instructions, as opposed to calculators such as card-programmed calculators on which the sequence is impressed manually.

Computer, Synchronous: A computer in which all operations and events are controlled by equally spaced pulses from a clock. Contrasted with *Computer, Asynchronous.*

Control: The computer circuits that affect the carrying out of instructions in the proper sequence, the interpretation of each instruction, and the application of the proper commands to other sections and circuits in accordance with the interpretation.

Control Word: (*See Word, Control.*)

Converter: A device that converts the representation of information or that permits the changing of the method of data processing from one form to another. An example is a unit, possibly including editing facilities, that accepts information from punched cards and records the information on magnetic tape.

Core Matrix: An array of cores, each of which represents the same column for each storage register in the magnetic-core storage system.

Core Storage: (*See Memory, Magnetic Core.*)

Counter: A device capable of increasing or decreasing its own contents upon receipt of separate input signals.

CPU: Central Processing Unit.

Data: A general term used to denote any or all numbers, letters, symbols, or facts that refer to or describe an object, idea, condition, situation, or other factors. The word *data* connotes basic elements of information that can be processed or produced by a computer. Sometimes data are considered to be expressible only in numerical form, but information is not so limited.

Data, Raw: Data that have not been processed. Such data may or may not be in machine-sensible form.

Debug: To isolate and remove all malfunctions from the computer, or all mistakes from a routine or program.

Decimal, Binary-Coded: (*See Binary-Coded Decimal.*)

Decision: Process to determine which of two or more alternate courses of action should be taken, based on comparison of available information.

Decoder: (1) A device that determines the meaning of a set of signals and initiates a computer operation based thereon. (2) A matrix of switching elements that selects one or more output channels according to the combination of input signals present.

Density, Character: The number of characters that can be stored per unit of length.

Digit: One of a set of characters used as coefficients of powers of the radix in the positional notation of numbers.

Disc, Magnetic: A storage device in which information is recorded on the magnetizable surface of a rotating disc. A magnetic-disc storage system is an array of such devices, with associated reading and writing heads mounted on movable arms. Related to *Memory, Disc.*

Drum, Magnetic: A cylinder having a surface coating of magnetic material, which stores binary information by the orientation of magnetic dipoles near or on its surface. Since the drum is rotated at a uniform rate, the information stored is available periodically as a given portion of the surface moves past one or more flux-detecting devices, called heads, located near the surface of the drum.

Dump: Transfer of information from one piece of equipment to another (normally from computer to external equipment such as paper tape, high-speed printer, etc.).

Enable: The application of a pulse that prepares a circuit for some subsequent action.

Encoder: A device capable of translating from one method of expression to another, e.g., translating the message, "Add the contents of A to the contents of B," into a series of binary digits.

EOF: End of File.

EOM: End of Message.

Erase: To replace all the binary digits in a storage device with binary zeros.

Exclusive OR Operator: A logical operator having the property that, if P and Q are two statements, then the statement P $\oplus$ Q (where $\oplus$ is the Exclusive OR operator) is true if either P or Q, but not both, is true, and false if P and Q are both false or both true.

Execution Time: The portion of an instruction cycle during which the actual work is performed or the operation executed, i.e., the time required to decode and perform an instruction.

Fault: (1) The condition resulting from the execution of an improper instruction. (2) A malfunction.

File: An organized collection of information directed toward some purpose. The records in a file may or may not be sequenced according to a key contained in each record.

Fixed-Point Arithmetic: (*See Arithmetic, Fixed-Point.*)

Fixed Word Length: The property that a machine word always contains the same number of characters or digits.

Flip-Flop: A bistable device, i.e., a device capable of assuming either of two stable states.

Floating-Point Arithmetic: (*See Arithmetic, Floating-Point.*)

Flowchart: A graphic representation of the major steps of work in process. The illustrative symbols may represent documents, machines, or actions taken during the process. Synonymous with process chart and *Flow Diagram.*

Flow Diagram: Same as *Flowchart.*

Format: The predetermined arrangement of characters, fields, lines, page numbers, and punctuation marks, usually on a single sheet or in a file. This refers to input, output, and files.

Function Code: The portion of the instruction word that specifies to the control section the particular instruction to be performed.

Gate, AND: A signal circuit with two or more input wires and one output wire. The output wire gives a signal if, and only if, all input wires receive coincident signals. Synonymous with *AND Circuit.*

Gate, OR: An electrical gate or mechanical device that implements the logical Inclusive OR operator.

Gray Code: A binary code in which sequential numbers are represented by expressions that are the same except in one place, and in that place differ by one unit.

Half Adder: A circuit having two output points, S and C, representing sum and carry, and two input points, A and B, representing addend and augend, such that the output is related to the input according to the following table:

Input		*Output*	
A	*B*	*S*	*C*
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

A and B are arbitrary input pulses, and S and C are sum without carry and carry, respectively. Two half adders properly connected may be used for performing binary addition and form a full serial adder.

Half-Subtract: The bit-by-bit subtraction of two binary numbers with no regard for borrows. Abbreviated HS.

Hardware: The physical equipment forming the digital system.

Inclusive OR Operator: A logical operator having the property that P + Q (where + is the Inclusive OR operator) is true if P or Q or both are true. When the term OR is used alone, as in "OR gate," the Inclusive OR is usually implied.

Input/Output: A section providing the means of communication between the computer and external equipment or other computers. Input and output operations involve units of external equipment, certain registers in the computer, and portions of the computer control section. Abbreviated i/o.

Interface: A common boundary between automatic data-processing systems or parts of a single system.

Interrupt: (1) Internal: Indicates the termination of an input or output buffer. (2) External: Signal on the data lines that requires computer attention.

I/O: Abbreviation for input/output.

Key: (1) A group of characters that identifies or is part of a record or item. Thus, any entry in a record or item can be used as a key for collating or sorting purposes. (2) A marked lever manually operated for copying a character, as on a typewriter, paper-tape perforator, card punch, digitizer, or manual word generator. (3) A lever or switch on a computer console for the purpose of manually altering computer action.

Keypunch: A device used to record information in cards or paper tape by punching holes to represent data.

Library: A collection of information available to a computer, usually on magnetic tapes.

Load: To enter information into either the computer or a storage location.

Logic: (1) The science dealing with the criteria or formal principles of reasoning and thought. (2) The systematic scheme that defines the interactions of signals in the design of an automatic data processing system.

Master Clock: The primary source of timing signals.

Memory: Same as *Storage*. A device in which data can be stored and from which it can be obtained at a later time. The means of storing data may be chemical, electrical, or mechanical.

Memory, Auxiliary: A storage device in addition to the main storage of a computer, e.g., magnetic tape, disc, or drum. Auxiliary storage usually holds much larger amounts of information than the main storage, and the information is accessible less rapidly.

Memory Capacity: (*See Capacity, Memory.*)

Memory, Disc: The storage of data on the surface of magnetic discs.

Memory, External: A facility or device, not an integral part of a computer, in which data usable by a computer are stored.

Memory, Internal: The storage facilities forming an integral physical part of the computer and directly controlled by the computer.

Memory, Magnetic: A device or devices that utilize the magnetic properties of materials to store information.

Memory, Magnetic Core: A storage device in which binary data are represented by the direction of magnetization in each unit of an array of magnetic elements. The elements are usually in the form of toroidal rings, but they may also be in other forms such as thin film. Synonymous with *Core Storage*.

Memory, Magnetic Drum: The storage of data on the surface of magnetic drums.

Memory, Magnetic Tape: A storage device in which data are stored in the form of magnetic areas on metal or coated plastic tape. Data bits are stored as small magnetized spots arranged in column form across the width of the tape. A read-write head is usually associated with each row of magnetized spots so that one column can be read or written at a time as the tape traverses the head.

Memory, Main: The fastest storage device of a computer, and the one from which instructions are executed. It is usually internal. Contrasted with *Memory, Auxiliary*.

Memory, Nonvolatile: A storage medium in which information is retained in the absence of power and may be made available upon restoration of power. Examples are magnetic tapes, cores, drums, and discs. Contrasted with *Memory, Volatile*.

Memory, Random-Access: A storage technique in which the time required to obtain information is independent of the location of the information most recently obtained. (This strict definition must be qualified by the observation that we usually mean "relatively random.")

Memory, Sequential-Access: A storage technique in which the items of information stored become available only in a one-after-the-other sequence, whether all the information or only some of it is desired.

Memory, Serial: A storage technique in which time is one of the factors used to locate any given bit, character, word, etc., and in which access time includes a variable latency, or waiting time, of from zero to many word times. Storage is said to be serial by word when the individual bits comprising a word appear serially in time. It is said to be serial by character when the characters representing coded decimal or other nonbinary numbers appear serially in time. For example, magnetic drums are usually serial by word but may be serial by bit, parallel by bit, or serial by character and parallel by bit.

Memory, Volatile: A storage medium in which information cannot be retained without continuous power dissipation. Contrasted with *Memory, Nonvolatile.*

Modulus: The number of permissible numbers used in a process or system. For example, if only the integers from -15 to $+15$ inclusive are considered, 31 is the modulus of this set of numbers.

Most Significant Digit: The first digit from the left different from zero.

Nondestructive Read: (*See Read, Nondestructive.*)

Nonvolatile Memory: (*See Memory, Nonvolatile.*)

Octal Number: A number in a system that uses eight symbols (0, 1, 2, 3, 4, 5, 6, and 7) and has eight as its base.

Off-Line: Descriptive of a mode of operation in which peripheral equipment is not under the control of the central processing unit.

On-Line: Descriptive of a mode of operation in which the peripheral equipment in a system is under control of the central processing unit, and in which information reflecting current activity is introduced into the data processing system as soon as it occurs.

Operator, AND: A logical operator that has the property that if P is a statement and Q is a statement, then P AND Q is true if both statements are true, but false if either is false or both are false. Truth is normally expressed by the value 1, falsity by 0. The AND operator is often represented by a centered dot ($P \cdot Q$), by no sign (PQ), by an inverted "u" or logical product symbol (PnQ), or by the letter x or multiplication symbol ($P \times Q$). Note that the letters AND are capitalized to differentiate between the logical operator AND and the word "and" in common usage.

Operator, OR: A logical operator that has the property that if P and Q are two statements, then the statement P OR Q is true if P, Q, or both are true, and false if both are false.

Overflow: The condition that arises when the result of an arithmetic operation exceeds the capacity of the storage space alloted in a digital computer.

Parity Bit: A check bit that indicates whether the total number of binary 1 digits in a character or word (excluding the parity bit) is odd or even. If a 1 parity bit indicates an odd number of 1 digits, then a 0 bit indicates an even number of them. If the total number of 1 bits, including the parity bit, is always even, the system is called an even-parity system. In an odd-parity system, the total number of 1 bits, including the parity bit, is always odd.

Partial Carry: A system of executing the carry process in which carries that arise as a result of a carry are not allowed to be transmitted to the next higher stage.

Peripheral Equipment: The auxiliary machines that may be placed under the control of the central computer. Examples of these are card readers, card punches, magnetic-tape feeds, and high-speed printers. Peripheral equipment may be used on-line or off-line depending on computer design, job requirements, and economics.

Positional Notation: A method for expressing a quantity by using two or more figures, which, taken successively from right to left, are to be interpreted as coefficients of ascending integral powers of the radix.

Radix: The base of a numbering system. The number of characters for use in each of the digital positions of the numbering system is equal to the radix. Unless otherwise indicated, the radix of any number is assumed to be 10. For positive identification, the radix may be written in parentheses as a subscript to the expressed number, e.g., $11_{(2)}$, $5_{(8)}$, or $126_{(10)}$. Synonymous with *Base*.

Radix-Minus-1 Complement: (*See Complement* [*on* $n - 1$].)

Rate, Bit: The rate at which binary digits, or pulses representing them, pass a given point in a communications line or channel.

Rate, Clock: The time rate at which pulses are emitted from the clock. The clock rate determines the rate at which logical or arithmetic gating is performed in a synchronous computer.

Rate, Sampling: The rate at which measurements of physical quantities are made. For example, if it is desired to calculate the velocity of a missile and its position is measured each millisecond, then the sampling rate is 1000 measurements per second.

Read-In: To sense information contained in some source and transmit this information to an internal storage.

Read, Nondestructive: A reading of the information in a register without changing that information.

Read Out: To sense information contained in some internal storage and transmit this information to a storage external to the computer.

Read Time: Same as *Access Time.*

Real-Time: Pertaining to computer operation with regard to a related process such that the computer results are available to conduct or guide the process.

Record: (1) A group of related facts or fields of information treated as a unit. Thus, a listing of information, usually in printed or printable form. (2) To put data into a storage device.

Register: A hardware device used to store a certain number of bits or characters. A register is usually constructed of elements such as transistors and usually contains approximately one word of information. Common programming usage demands that a register have the ability to operate on information and not merely store it; hardware usage does not make this distinction.

Register, Storage: A register in the storage of a computer, in contrast with a register in one of the other units of the computer.

Sampling Rate: (*See Rate, Sampling.*)

Scale: A range of values frequently dictated by the computer word length or routine at hand.

Sense: (1) To determine the present arrangement of some element of hardware, especially a manually set switch. (2) To read punched holes or other marks.

Sensitivity: The degree of response of an instrument or control unit to a change in the incoming signal.

Serial: The handling of entities one after the other in a single facility, such as to transfer or store in a digit-by-digit time sequence or to process a sequence of instructions one at a time.

Serial-Parallel: (1) A combination of serial and parallel, such as serial by character and parallel by bits comprising the character. (2) Descriptive of a device that converts a serial input into a parallel output.

Shift: To move the characters of a unit of information columnwise right or left. For a number, this is equivalent to multiplying or dividing by a power of the base of notation.

Shift, Arithmetic: To multiply or divide a quantity by a power of the number base. For example, if binary 1101, which represents decimal 13, is arithmetically shifted twice to the left, the result is 110100. This represents 52, which is also obtained by multiplying 13 by 2 twice. By comparison, if decimal 13 were to be shifted to the left twice, the result would be the same as multiplying by 10 twice, or 1300.

Shift Register: A register in which the characters may be shifted one or more positions to the right or left.

Sign Digit: A binary digit used as a sign.

Software: The totality of programs and routines used to extend the capabilities of computers, such as compilers, assemblers, narrators, routines, and subroutines. Contrasted with *Hardware.*

Sorter: A machine that puts items of information into a particular order; e.g., it will determine whether A is greater than, equal to, or less than B and will then sort or order accordingly.

Storage: (*See Memory.*)

Switch, Toggle: (1) An electronically operated circuit that holds either of two states until changed. (2) A manually operated electric switch, with a small projecting knob or arm that may be placed in either of two positions, "on" or "off," and will remain in that position until changed.

Table: A collection of data in a form suitable for ready reference, frequently as stored in sequenced machine locations or written in the form of an array of rows and columns for easy entry and in which an intersection of labeled rows and columns serves to locate a specific piece of data or information.

Table, Truth: A representation of a switching function, or truth function, in which every possible configuration of argument values 0 and 1 or true and false is listed, and beside each is given the associated function value (0 or 1, or true or false). The number of configurations is 2^n, where n is the number of arguments.

Ternary: Pertaining to a system of notation utilizing the base 3.

Test, Diagnostic: The running of a machine program or routine for the purpose of discovering a failure or potential failure of a machine element, and to determine its location or potential location.

Time, Execution: (*See Execution Time.*)

Time, Instruction: (1) The portion of an instruction cycle during which the control unit is analyzing the instruction and setting up to perform the indicated operation. (2) Same as *Execution Time.*

Time, Real (*See Real-Time.*)

Time Sharing: The use of a device for two or more purposes during the same overall time interval, accomplished by interspersing component actions in time.

Toggle: (1) A flip-flop. (2) Pertaining to a manually operated on-off switch, i.e., a two-position switch. (3) Pertaining to flip-flop, seesaw, or bistable action.

Transmission, Serial: To move data in sequence, one character at a time, as contrasted with parallel transmission.

Transport, Tape: The mechanism that moves magnetic or paper tape past sensing and recording heads. It is usually associated with data-processing equipment. Synonymous with tape transport, tape drive, and tape feed.

Truth Table: (*See Table, Truth.*)

Update: (1) To put into a master file changes required by current information or transactions. (2) To modify an instruction so that the address numbers it contains are increased by a stated amount each time the instruction is performed.

Variable: (1) A quantity that can assume any of the numbers of some set of numbers. (2) A condition, transaction, or event that changes or may be changed as a result of processing additional data through the system.

Variable Word Length (*See Word Length, Variable.*)

Vector: A quantity having magnitude and direction, as contrasted with a scalar, which has magnitude only.

Word: An ordered set of characters that occupies one storage location and is treated by the computer circuits as a unit and transferred as such. Ordinarily, a word is treated as an instruction by the control unit and as a quantity by the arithmetic unit. Word lengths may be fixed or variable, depending on the particular computer.

Word, Control: A word, usually the first or last of a record or block, which carries indicative information for the following words, records, or blocks.

Word, Data: A word that may be primarily regarded as part of the information manipulated by a given program. A data word may be used to modify a program instruction, or it may be arithmetically combined with other data words.

Word Length, Fixed: (*See Fixed Word Length.*)

Word Length, Variable: Having the property that a machine word may have a variable number of characters. It may be applied either to a single entry whose information content may be changed from time to time, or to a group of functionally similar entires whose corresponding components are of different lengths.

Write: (1) To transfer information, usually from main storage, to an output device. (2) To record data in a register, location, or other storage device or medium.

Zero Address Instruction: An instruction consisting of an operation which does not require the designation of an address in the usual sense. For example, the instruction, "Shift left 0003," has in its normal address position the amount of the shift desired.

Zone: (1) A portion of internal storage allocated for a particular function or purpose. (2) The three top positions of 12, 11, and 10 on certain punch cards. In these positions, a second punch can be inserted so that with punches in the remaining positions 1 to 9, alphabetic characters may be represented.

APPENDIX **B**

Answers to Exercises

CHAPTER 1

A1-1. (A) A bit is a *bi*nary digi*t*, 0 or 1.
(B) A group of bits composing a character.
(C) A number, letter, or symbol represented by a byte.
(D) An ordered set of characters in one storage location treated by the computer as a unit and transferred as such. A word may be treated by a control unit as an *instruction*, and by an arithmetic unit as a *quantity*. Word lengths may be fixed or variable depending on the particular system. In general, words are specified by the manufacturer.
(E) Baud is the reciprocal of the shortest signal element.

A1-2. (A) Bauds = 1/0.001 = 1000
(B) Total time required:

$$6(0.001) = 0.006 \text{ s}$$
$$\text{Addition of stop pulse } \underline{0.004 \text{ s}}$$
$$\text{Total} = \overline{0.010 \text{ s}} = 10 \text{ ms}$$

Then:

$$\text{Information bits/s} = (1/0.01)(5)$$
$$= 100(5)$$
$$= 500 \text{ information bits/s}$$

A1-3. (A) Bauds = 1/0.001 = 1000 (no change)
(B) Total time: $6(0.001) = 0.006$ s
Stop bit = 0.002 s
Total = 0.008 s (8 ms)
Information bits/second:

$$(1/0.008)(5) = 125\ (5)$$
$$= 625 \text{ (increased by 125)}$$

A1-4. There is a total of seven pulses in 8 ms, so:

$$(1/0.008)(7) = (125)(7) = 875 \text{ bits/s}$$

A1-5. (A) Multiply hartleys by 3.3 to get bits/second.
(B) Divide bits/second by 3.3 to get hartleys.

A1-6. Note that 50 dB is a power ratio of 10^5. Then:

$$\begin{aligned} C &= 5000 \log_{10} (1 + 10^5) \\ &= 5000 \log (10^5) \\ &= 5000 (5) \\ &= 25{,}000 \text{ hartleys} \end{aligned}$$

Then:

$25{,}000 \times 3.3 = 82{,}500$ bits/second absolute maximum

A1-7. $82{,}500/8 = 10{,}312$ characters/second maximum

A1-8. Since C (in hartleys) $= B \log_{10} (1 + s/n)$, then:

$$B = \frac{C \text{ (hartleys)}}{\log_{10} (1 + s/n)}$$

Convert bits/second to hartleys:

$$82{,}500/3.3 = 25{,}000$$

Then:

$$B = \frac{25{,}000}{\log 1000}$$

$$= \frac{25{,}000}{3} = 8333 \text{ Hz min bandwidth}$$

A1-9. Convert bits/second to hartleys:

$$100(10^6)/3.3 = 3.03(10^7) \text{ hartleys}$$

Then:

$$\begin{aligned} B &= 3.03(10^7)/\log 10^5 \\ &= 3.03(10^7)/5 \\ &= 6.06(10^6) \text{ hertz} = 6.06 \text{ MHz minimum} \end{aligned}$$

A1-10. Yes:

$$\begin{aligned} C \text{(in hartleys)} &= B \log_{10} (1 + s/n) \\ &= 2500 \log_{10} (1 + 1) \\ &= 2500 \log_{10}(2) \\ &= 2500(0.3010) \\ &= 752 \text{ hartleys} \end{aligned}$$

Then:

$$752 \times 3.3 = 2481 \text{ bits/s}$$

NOTE: Example 10 points out that even a very noisy channel can be utilized provided the message is stretched out, which simply means a lower transmission rate.

Compare the capacity for a 1/1 signal-to-noise ratio (2481 bits/s) with that obtained for the same bandwidth as solved in Section 1-4 for a 30-dB s/n ratio. This was 24,750 bits/s, a marked increase in capacity.

Remember also that for the 1/1 signal-to-noise ratio the 2481 bits/s would include all parity bits or bits with redundant message enhancement. This is true for *any* system with parameters of capacity (in bits/second), bandwidth (in hertz), and s/n power ratio.

In practice, other channel imperfections and limitations of the equipment impose a minimum desirable signal-to-noise power ratio of 100/1, or 20 dB.

CHAPTER 2

A2-1. (A) $32 + 8 + 1 + 0.25 = 41.25$
(B) $8 + 2 + 1 + 0.5 + 0.25 + 0.125 + 0.0625 = 11.9375$
(C) $128 + 64 + 32 + 8 + 2 + 0.5 + 0.125 = 234.625$
(D) $1 + 0.0625 = 1.0625$

A2-2. (A)
$$\begin{array}{rll} & 1000 & (8) \\ + & 1001 & (9) \\ \hline & 10001 & (17) \end{array}$$

(B)
$$\begin{array}{rll} & 1101.100 & (13.5) \\ + & 1000.101 & (8.625) \\ \hline & 10110.001 & (22.125) \end{array}$$

(C)
$$\begin{array}{rll} & 0000.1110 & (0.875) \\ + & 0001.1110 & (1.875) \\ \hline & 0010.1100 & (2.75) \end{array}$$

(D)
$$\begin{array}{rll} & 0000.11111 & (0.96875) \\ + & 0011.11100 & (3.875) \\ \hline & 0100.11011 & (4.84375) \end{array}$$

Note: $2^{-5} = 0.03125$

A2-3. (A) Arithmetical:
$$\begin{array}{rl} & 1110 \\ - & 0110 \\ \hline & 1000 \end{array}$$

Complement:
$$\begin{array}{rl} & 1001 \quad \text{(Ones complement)} \\ + & \quad 1 \\ \hline & 1010 \\ \text{add} & 1110 \\ \hline & 11000 \quad \text{(Answer)} = +8 \end{array}$$

(B) Arithmetical:
$$\begin{array}{rl} & 101001 \\ - & 001010 \\ \hline & 011111 \quad \text{(Answer)} \end{array}$$

The procedure for the above is as follows, starting with the LSD as the first digit:

1st digit: $1 - 0 = 1$.
2nd digit: $0 - 1 =$ difference of 1 and borrow 1.
3rd digit: No 1 exists to borrow yet, so 0 in the top row becomes 1, and $1 - 0 = 1$ with borrow 1 carried over.
4th digit: Now a 1 exists to borrow, so 1 in the top row becomes 0, and $0 - 1 =$ difference of 1 and borrow 1.
5th digit: No 1 exists to borrow, so 0 in the top row becomes 1, and $1 - 0 = 1$ with borrow 1 carried over.

6th digit: Now a 1 exists to borrow, so 1 in the top row becomes 0, and $0 - 0 = 0$. Prove the answer by noting that the decimal equivalent is $41 - 10 = 31$. Note that this procedure is cumbersome relative to the following complementary operation:

(B) Complement:

$$\begin{array}{rr} & 110101 \quad \text{(Ones complement)} \\ + & \underline{\quad\quad 1} \\ & 110110 \quad \text{(Twos complement)} \\ \text{add} & \underline{101001} \\ & 1011111 = +011111 = 31 \end{array}$$

A2-4. (A) See Fig. B-1A for answer.
(B) See Fig. B-1B for answer.
(C) See Fig. B-1C for answer.

A2-5. (A) Straight arithmetical division:

$$\begin{array}{r} 0011 \quad \text{(Answer)} \\ 1100\overline{)100100} \\ \underline{1100} \\ 1100 \\ \underline{1100} \\ 0000 \end{array}$$

Accumulator	Multiplier	Operation
00000000	1110	Move multiplicand 1 space left.
1011-		Add.
10110	1100	Move multiplicand 2 spaces left.
1011--		Add.
1000010	1000	Move multiplicand 3 spaces left.
1011---		Add.
10011010	0000	Answer = 154 (11 x 14 = 154)

(A) *1011 × 1110.*

Accumulator	Multiplier	Operation
000000000	11010	Move multiplicand 1 space left.
1011-		Add.
10110	11000	Move multiplicand 3 spaces left.
1011---		Add.
1101110	10000	Move multiplicand 4 spaces left.
1011----		Add.
100011110	00000	Answer = 286 (11 x 26 = 286)

(B) *1011 × 11010.*

Accumulator	Multiplier	Operation
00000000	1111	
1111		Add.
1111	1110	Move multiplicand 1 space left.
1111-		Add.
101101	1100	Move multiplicand 2 spaces left.
1111--		Add.
1101001	1000	Move multiplicand 3 spaces left.
1111---		Add.
11100001	0000	Answer = 225 (15 x 15 = 225)

(C) *1111 × 1111.*

Fig. B-1. Answers to question Q2-4.

Over-and-over subtraction:

```
  100100
- 1100
-------
  11000   First subtraction
- 1100
-------
  1100    Second subtraction
- 1100
-------
  0000    Third subtraction
```

Answer is decimal 3 = binary 0011

(B) Straight arithmetical division:

```
          0100   (Answer)
      ----------
0010 / 1000
       10
       --
       0000
```

Over-and-over subtraction:

```
  1000
-0010
-----
  0110   First subtraction
-0010
-----
  0100   Second subtraction
-0010
-----
  0010   Third subtraction
-0010
-----
  0000   Fourth subtraction
```

Answer is decimal 4 = binary 0100

A2-6. See Fig. B-2 for answers.

A2-7. (A) 1000 0110 0100. 0010
(B) 0001 0000 1000 0111
(C) 0010 0101. 1000
(D) 0010 0110 0101

A2-8. (A) 001 000 110 100 binary 1000110100
1 0 6 4 = octal 1064

(B) 111 100 100 . 010 binary 111100100.01
7 4 4 . 2 = octal 744.2

	256	128	64	32	16	8	4	2	1	.	0.5	0.25	0.125	0.0625	.03125
(A)			1	1	1	1	1	0	0	.	1	1	1		
(B)	1	0	0	0	0	0	1	1	0	.					
(C)									0	.	1	0	0	0	1
(D)		1	0	1	1	1	0	1	1	.	1	0	1		

Fig. B-2. Answers to question Q2-6.

A2-9. (A) Octal: 7 4 4 . 2

4-2-1 binary: $\overbrace{111}\ \overbrace{100}\ \overbrace{100}\ .\ \overbrace{010}$

Pure binary: 111100100.010

$256 + 128 + 64 + 32 + 4 + 0.25 = 484.25$ decimal

(B) Octal: 1 0 6 4

4-2-1 binary: $\overbrace{001}\ \overbrace{000}\ \overbrace{110}\ \overbrace{100}$

Pure binary: 001000110100

$512 + 32 + 16 + 4 = 564$ decimal

(C) Octal: 2 7 2 . 5

4-2-1 binary: $\overbrace{010}\ \overbrace{111}\ \overbrace{010}\ .\ \overbrace{101}$

Pure binary: 010111010.101

$128 + 32 + 16 + 8 + 2 + 0.5 + 0.125 = 186.625$ decimal

(D) Octal: 3 1 . 7

4-2-1 binary: $\overbrace{011}\ \overbrace{001}\ .\ \overbrace{111}$

Pure binary: 011001.111

$16 + 8 + 1 + 0.5 + 0.25 + 0.125 = 25.875$ decimal

A2-10. (A) Octal $31.1 = (3 \times 8^1) + (1 \times 8^0) + (1 \times 8^{-1})$

$= (3 \times 8) + (1 \times 1) + (1 \times 0.125)$

$= 24 + 1 + 0.125 = 25.125$ decimal

(B) Octal $1064 = 512 + 48 + 4 = 564$ decimal

A2-11. (A) 110100 (Inverted 001011) Ones complement

$+ \quad 1$

110101 (Complement of 001011) Twos complement

Add 110101

$\overline{1101010} = +42$ decimal

(B) 00101 (Inverted 11010) Ones complement

$+ \quad 1$

00110 (Complement of 11010) Twos complement

Add 1111

$\overline{010101} = -10101$

Recomplement: 01010

$+ \quad 1$

$\overline{01011} = -1011 = -11$ decimal

(C) 0000101 (Inverted 1111010) Ones complement

$+ \quad 1$

0000110 (Complement of 1111010)

Twos complement

Add 1110101

$\overline{01111011} = -1111011$

Recomplement: 0000100 (Inverted)

$+ \quad 1$

$\overline{0101} = -0101 = -5$ decimal

(D)
$$\begin{array}{rrl} 1110 & 1101 & \text{(Inverted subtrahend)} \\ & & \text{Ones complement} \\ +\ \ 1 & +\ \ 1 & \\ \hline 1111 & 1110 & \text{(Complement of subtrahend)} \\ & & \text{Twos complement} \\ \text{Add } 1001 & 0100 & \\ \hline 11000 & 10010 & = \text{bcd } 1000 \quad 0010 = 82 \text{ decimal} \end{array}$$

(E)
$$\begin{array}{rrl} 0110 & 1011 & \text{(Inverted subtrahend)} \\ & & \text{Ones complement} \\ +\ \ 1 & +\ \ 1 & \\ \hline 0111 & 1100 & \text{(Complement of subtrahend)} \\ & & \text{Twos complement} \\ \text{Add } 0001 & 0010 & \\ \hline 01000 & 01110 & = -(1000\ \ 1110) \end{array}$$

$$\begin{array}{rrl} \text{Recomplement: } 0111 & 0001 & \text{(Inverted)} \\ +\ \ 1 & \ \ 1 & \\ \hline 1000 & 0010 & \end{array}$$

Recomplemented $= -(1000\ \ 0010) = -82$ decimal

Note that the complementary methods of computation are the same for bcd as for pure binary. Just remember to add 1 to *each* group of four bits.

A2-12. (A) $2^4 = 16$. (Note that 2^3 is not sufficient.)
(B) $2^7 = 128$. (Note that 2^6 is not sufficient.)
(C) $2^{10} = 1024$.
(D) $2^{14} = 16{,}384$.
(E) $2^{15} = 32{,}768$.

Remember that the exponent of 2 is the number of bits used in the system. Thus, 8 bits (2^8) allow 256 possible characters or levels.

CHAPTER 3

A3-1. (A) "A AND B." (B) "NOT A AND B" (inverted AB). (C) "A NOT AND B NOT." (D) "A OR B Inclusive" (either or both). (E) "NOT A OR B" (inverted A + B). (F) A OR B Exclusive (either/or, but not both).

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

Fig. B-3. Answer to question Q3-2.

A	B	C	D	f
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Fig. B-4. Answer to question Q3-3.

A3-2. See Fig. B-3. This is a NOR circuit preceded by NOTS. Note that when you have inverted inputs to a NOR gate, the combination becomes an AND gate.

A3-3. See Fig. B-4. This is a four-input AND gate. Note that the truth table starts with binary 0000 and increases by adding 1 each time until the number 15 (1111) is reached. This assures that no combination is omitted.

A3-4. (A) This is NOR logic. See Fig. B-5A for the truth table.
(B) The equivalent schematic symbol is shown in Fig. B-5B.

A3-5. (A) $AC + ABC + A\overline{C}$

$= A(C + \overline{C}) + ABC$ (Factoring and rearranging)

$= A(1) + ABC$ (Rule 12)

$= A + ABC$ (Rule 16)

$= A$ (Rule 22)

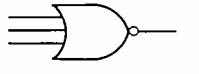
(B) See Fig. B-6 for solution by truth table. Review Section 3-8 for method of construction.

This result means that a straight wire from A is equivalent to $AC + ABC + A\overline{C}$.

A3-6. See Fig. B-7. This is the logic schematic for a full adder, where C = previous carry and C_o = new carry output.

A	B	C	$f = \overline{A+B+C}$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

(A) *Truth table.*



(B) *Logic symbol.*

Fig. B-5. Answers to question Q3-4.

A	B	C	$\bar{C}$	AC	$A\bar{C}$	ABC	$AC+ABC+A\bar{C}$
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	0	0	0	0	0
1	0	0	1	0	1	0	1
1	0	1	0	1	0	0	1
1	1	0	1	0	1	0	1
1	1	1	0	1	0	1	1
Basic				Basic Combinations Called for in Boolean Relationship			Combination of Combinations

Fig. B-6. Answer to question Q3-5B.

A3-7. Case 1: $f = \overline{A}\overline{B}$
Case 2: $f = \overline{A + B}$
Explanation:
Case 1: A negative voltage (logical 0) on either or both diodes causes Q1 to conduct (saturate), sending f to essentially ground (logical 1). Transistor Q1 is cut off *only* when *both* inputs are at ground (logical 1), sending f to $-V_{CC}$ (logical 0). Note the polarity inversion. Hence, the output is AB with polarity inversion, or $\overline{AB}$. This is a positive-logic NAND gate.
Case 2: The physical explanation is the same as for Case 1, but with logic-level reversal. A negative voltage (logical 1) on either or both diodes causes Q1 to saturate (closed switch to ground), sending f to essentially ground potential (logical 0). Transistor Q1 is cut off only when both inputs are at ground (logical 0), sending f to $-V_{CC}$ (logical 1). Note the polarity inversion. Hence, the output is $A + B$ with polarity inversion, or $\overline{A + B}$. This is a negative-logic NOR gate.

A3-8. Fan-out is the capability of a given device to supply current to and sink current from other devices. It is generally given as the number (n) of inputs that can be connected to an output.

CHAPTER 4

A4-1. By tying J and K together and applying the trigger pulse to the common input.

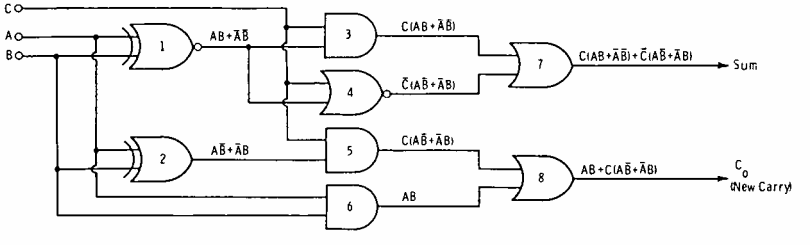


Fig. B-7. Answer to question Q3-6.

A4-2. The JK flip-flop has no indeterminate outputs.

A4-3. Data (0 or 1) may be stored by the flip-flop and recalled at a later time by application of the appropriate pulse.

A4-4. For the pulse-triggered flip-flop, data stored *prior* to the first clock transition are recognized. For the edge-triggered type, data occurring immediately *after* the clock transition are stored and are released on the trailing edge of the clock pulse.

A4-5. No. A reset pulse can be made to coincide with the binary equivalent of an odd number, and this pulse can be used to restart the count to obtain division by an odd integer.

A4-6. A register in which binary characters may be shifted one or more positions to the left or right.

A4-7. The decoder/driver for a single cathode that represents a given character provides only one output for a given binary input. The decoder/driver for a segmented display transmits as many outputs as necessary for the segments composing the character.

A4-8. This means that all display devices include a decimal point, which is driven from an external control circuit. Thus, the decimal point may be placed anywhere in the number sequence. This contrasts with the fixed decimal display, in which only one display device (fixed in the number sequence) includes the decimal point, which is always illuminated for the number in that position.

CHAPTER 5

A5-1. (A) Weighted.
(B) None. Therefore, this is an arbitrarily weighted code.

A5-2. There are always exactly two 1's in every binary word. This is an error-detecting code (detects a single error).

A5-3. See Table B-1.

A5-4. Even parity indicates that the total number of 1's, including the parity bit, for any binary word is an even number. Odd parity indicates that the total number of 1's, including the parity bit, in a binary word is an odd number. Odd parity is preferred for any binary word that has all zeros for decimal 0, so that a 1 appears in the word. When zeros indicate no pulses, this differentiates between a total lack of information and an actual binary word.

A5-5. (A) The bcd code employs a 4-bit word to represent the numbers 0 to 9 (ten numbers). But $2^4 = 16$, so there are six redundant binary-coded decimal digits (not used). The redundancy relationship becomes:

Table B-1. Solution to Q5-3

Decimal Digit	7	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	1	0	0	0
8	1	0	0	1
9	1	0	1	0

$$\text{Redundancy} = 1 - \frac{\text{Information digits}}{\text{Total digits}}$$

$$= 1 - \frac{10}{16} = 1 - 0.625 = 0.375 = 37.5\%$$

(B) The efficiency of the bcd code is $1 - 0.375 = 0.625 = 62.5\%$. Or:

$$E = \frac{10}{16} = 0.625 = 62.5\%$$

A5-6. $E = \frac{4}{4} = 1 = 100\%$

A5-7. $E = \frac{4}{5} = 0.8 = 80\%$

A5-8. The total number of bits would be 28; therefore:

$$E = \frac{14}{28} = 0.5 = 50\%$$

CHAPTER 6

A6-1. The sense wire serves as the output line. It picks up a small voltage when the flux changes from the 1 state to the 0 state.

A6-2. In a memory system, the output often feeds one input of an AND circuit. The other input receives a "strobe" pulse, which is timed to occur at the midpoint of the output pulse. Thus, the output from the AND circuit is "alive" only at the point of maximum output-pulse intensity, minimizing any effects of noise. In this manner, the strobe pulse is said to "sample" a circuit to see if infor-

mation is present. If data are there, they are passed along; if no data are present, an "open circuit" exists.

A6-3. Each core is one bit. Therefore:

$$16 \times 16 = 256 \text{ Bit capacity}$$

A6-4. Let n = Number of 1-bit words
m = Number of bits per word

In magnetic-core memory, a capacity of 4096 words by 8 bits would require eight planes of 4096-core arrays. Thus, n = 4096 and m = 8.

For IC memory, assume each chip has 256 words. In this case, a capacity of 4096 words by 8 bits would require 4096/256 = 16 chips in each of eight planes. Thus: n = 16 and m = 8. The total number of ICs required would be m × n, or 16 × 8 = 128.

A6-5. (A) The abbreviation dro means "destructive readout." For example, a magnetic core in the 1 state is returned to the 0 state upon readout.

(B) The abbreviation ndro means nondestructive readout. For a magnetic core, this can be achieved by immediately rewriting the information following readout. Integrated circuits are inherently ndro's, as are magnetic discs, cards, tape, etc.

(C) A storage medium in which information cannot be retained without continuous power dissipation.

(D) A storage medium in which information is retained in the absence of power and may be made available upon restoration of power.

A6-6. It is necessary to erase for normal analog recording or for RZ digital data. It is not necessary to erase for NRZ or RB digital methods of recording.

A6-7. The main factors are quality of the magnetic coating, closeness of the head to the coating, and the narrowness of the head gap.

A6-8. Often they do not. Most peripheral devices use characters containing 5 to 10 bits. Many basic digital units contain words of from 12 to 36 bits.

A6-9. At 3600 rpm:

Revolutions per second = 3600/60 = 60
1 revolution = 1/60 = 0.0167 second = 16.7 milliseconds

This is the latency, or rotational delay. Since the maximum access time is the latency plus the seek time:

$$0.0167 + 0.1 = 0.1167 \text{ s} = 117 \text{ ms}$$

A6-10. Magnetic tape is quite satisfactory for sequential programming. However, if random access is important, any magnetic card can be reached as quickly as any other, decreasing the access time. A whole reel of magnetic tape would have to be rewound if the desired data happened to be at the far end of the tape.

CHAPTER 7

A7-1. (A) Adc means "analog-to-digital converter."
(B) Dac means "digital-to-analog converter."
(C) Codec means a combination encoder-decoder.

A7-2. 1/256 volt = 0.0039 volt = 3.9 millivolts

A7-3. $2^{10} = 1024$. 1023 comparators would be required.

A7-4. Direct comparison is a "one-look" (simultaneous) converter; hence, it is very fast in operation. However, it requires a large number of converters ($2^n - 1$, where n is the number of bits in the binary word).

The multiple-comparison subranging adc is almost as fast, but it requires much less hardware. It is much faster than the successive-approximation technique.

A7-5. (A) A unit-distance code, such as the Gray code (Section 7-8), has only a 1-bit change in value from one position to the next higher or lower position. Thus, the error in readout of adjacent sectors cannot exceed 1.

(B) It is normally employed for mechanical applications such as converting the positions of a rotating shaft to digital data.

A7-6. (A)

$$E_o = \frac{I(500)}{3}\left(\frac{1000}{1000 + 1000/3}\right)\frac{6}{2}$$

$$= \frac{I(500)}{3}\left(\frac{1000}{1333}\right)3$$

$$= 167I\ (0.75)\ 3$$

$$= 167I\ (2.25)$$

$$= 375I$$

Then for I = 5 mA:

$$E_o = 375(0.005) = 1.88 \text{ volts}$$

(B)

$$E_o = 167I\ (0.75)\ (3/2)$$

$$= 167I\ (0.75)\ (1.5)$$

$$= 167I\ (1.125)$$

$$= 188I$$

Then for I = 5 mA:

$$E_o = 188(0.005) = 0.94 \text{ volt}$$

Note that the output for decimal 3 is exactly half the output for decimal 6.

CHAPTER 8

A8-1. A complete system of equipment necessary to allow automatic transmission of data between two or more points.

A8-2. (A) On-line operation permits two or more digital units to communicate directly with each other.

(B) Off-line operation means that the output of one or more digital units is fed into a memory (storage) device. This stored data can then be fed into a second digital unit, or back into the first one, at a more convenient time.

A8-3. (1) Baseband, (2) modulated baseband, (3) modulated carrier.

A8-4. (1) Simplex (one-way only), (2) half-duplex (two-way but not simultaneously), (3) full duplex (two-way simultaneously).

A8-5. The binary form of letters, numbers, audio, video, etc.

A8-6. (1) To move dc and low-frequency ac components above the low-frequency cutoff of a transmission path. (2) To provide more efficient use of available bandwidth.

A8-7. Review Section 8-3. The equation for the minimum sampling rate (f_s) is:

$$f_s = 2B(1 + k/m)$$

The values to substitute in this equation are:

$B = f2 - f1 = 32\text{ kHz} - 29\text{ kHz} = 3\text{ kHz}$

$m = f2/B = 32/3 = 10.66 = 10$ (Largest integer)

$k = f2/B - m = 32/3 - 10 = 10.66 - 10 = 0.66$

Then

$$\begin{aligned} f_s &= 2 \times 3\text{ kHz}\,(1 + 0.66/10) \\ &= 6\text{ kHz}(1 + 0.066) \\ &= 6\text{ kHz}(1.066) \\ &= 6.4\text{ kHz} \end{aligned}$$

A8-8. Frequency-division multiplexing (fdm) and time-division multiplexing (tdm).

A8-9. (A)

$$\text{Minimum } f_s = 2f_h = 2 \times 10 = 20\text{ Hz}$$

$$\text{Maximum frame time} = 1/f_s = 1/20 = 0.05\text{ s} = 50\text{ ms}$$

(B) For ten channel pulses with equal spaces:

$$\text{Max pulse time} = \frac{\text{Max frame time}}{\text{No of pulses} + \text{No of spaces}}$$

$$= \frac{0.05}{10 + 10} = \frac{0.05}{20} = 0.0025\text{ s} = 2.5\text{ ms}$$

A8-10. To provide a flat-topped pulse that allows the time necessary to convert the instantaneously sampled level to a binary code.

CHAPTER 9

A9-1. Partially. Digital control means an analog function is *controlled* by digital information without conversion of analog form to

digital form. However, dc values representing a given operating parameter may be converted to a digital code.

A9-2. A unique digital *code* is used for each control parameter.

A9-3. $2^5 = 32$ functions.

A9-4. One bit ($2^1 = 2$ functions).

A9-5. By modulating a carrier, usually around 1200 to 1500 Hz.

A9-6. By a voltage-controlled oscillator (vco) whose frequency is dependent on a dc value. When the dc value changes, the output frequency changes accordingly.

A9-7. Phase-shift keying in which a 180° phase reversal of the fixed tone frequency occurs at the leading and trailing edges of a pulse.

A9-8. The number of "dots" employed to make up each character.

A9-9. The character resolution, or number of dots available for a single character.

A9-10. A computer system that enables the operator to "practice" a complete editing procedure by watching a picture monitor (and listening to an audio monitor) without disturbing the original recording. Then, when the entire procedure is satisfactory, the information is stored again in the computer, and a single push button initiates automatic editing.

CHAPTER 10

A10-1. (1) Temporary storage or (2) time multiplexing of signals.

A10-2. Once placed in digital storage or memory, it can be processed in any number of ways and read out on any desired time base.

A10-3. To minimize "aperture" distortion, caused by rapid signal-level changes between samples.

A10-4. To prevent passage to the adc of any signal or transient response that approaches or exceeds one-half the sampling frequency.

A10-5. To prevent switching transients above the intended passband from appearing at the output.

A10-6. To prevent large "glitches" in the audio or video output.

A10-7. The minimum sampling frequency is normally at least 10% greater than twice the highest frequency in the intended passband.

A10-8. Maximum. (Review Section 10-1.)

A10-9. Review Section 10-1. Let B = bandwidth:

$$B = \frac{100(10^6)}{\log_2 10{,}000}$$

$$= \frac{100{,}000{,}000}{13.2}$$

$$= 7.6 \text{ MHz}$$

Note that if the original analog video signal has the usual bandwidth of about 4 MHz, then about twice the baseband frequency bandwidth is required for digitized video.

A10-10. The compander compresses data signals by acting on the number of bits forming a code word according to a prescribed relationship for a given system.

A10-11. No. Information rate is determined by the code efficiency factor. (Review Section 5-7.)

A10-12. The level (usually given in decibels) below an overload, limiting, or specified distortion level.

A10-13. No. A RAM is a random-access memory. A shift register is a SAM, or serial-access memory.

A10-14. The phase of the sampling frequency is shifted 180° on alternate horizontal scan lines. The PALE technique is used in systems with 10.7-MHz sampling rates (3 × color-subcarrier frequency) to allow efficient use of comb filters in decoding. It is not necessary in systems employing 4 × color-subcarrier frequency (14.3 MHz).

A10-15. One field, or more than 1.5 million bits for the most efficient codes.

CHAPTER 11

A11-1. The logic probe allows clock and other inputs to be observed and reveals pulse activity at the output. Timing of the output is generally correct, since if a failure occurs in the digital world, it is usually a catastrophic failure: the output will have no pulse activity and will be stuck high or low, as revealed by the logic probe.

A11-2. Under normal operation for a given mode, the inputs to the NAND gate are high, low, and low and are so indicated by the logic probe. The output should then be high. If an error must occur to send the output low (which means all inputs must be high), the logic pulser can be used to pulse the low pins high (by a multipin kit), and the output can be checked with the logic probe. The output should now be low.

A11-3. Use a logic pulser and a logic probe simultaneously on the same pin. While the logic pulser is powerful enough to override a low-impedance TTL output, it is not powerful enough to effect a change in state on a V_{CC} or ground bus. Thus, if a logic pulser is used to inject a pulse while the logic probe is used simulta-

neously on the same pin to observe the pulse, a short to V_{CC} or ground can be detected. Occurrence of a pulse indicates that the node is not shorted, and the absence of a pulse indicates the node is shorted to V_{CC} (if it is high, or steady bright glow) or ground (if lamp is extinguished).

A11-4. (1) Short in the printed-circuit wiring path. (2) Internal short in the output or input of a driver or driven IC. (3) Short in any analog component such as resistor, capacitor, etc., that might be attached to the node.

A11-5. The logic probe will show "bad level" (dim light) at the input pin. The open input will be interpreted as a high level in determining the output of the logic IC.

A11-6. The logic clip.

A11-7. When a pulse-memory option is used, monitoring of a signal line for single-shot or low-frequency pulses over extended periods of time is possible. An LED in the pulse memory will light upon the occurrence of a pulse, and it will remain on until reset by the user.

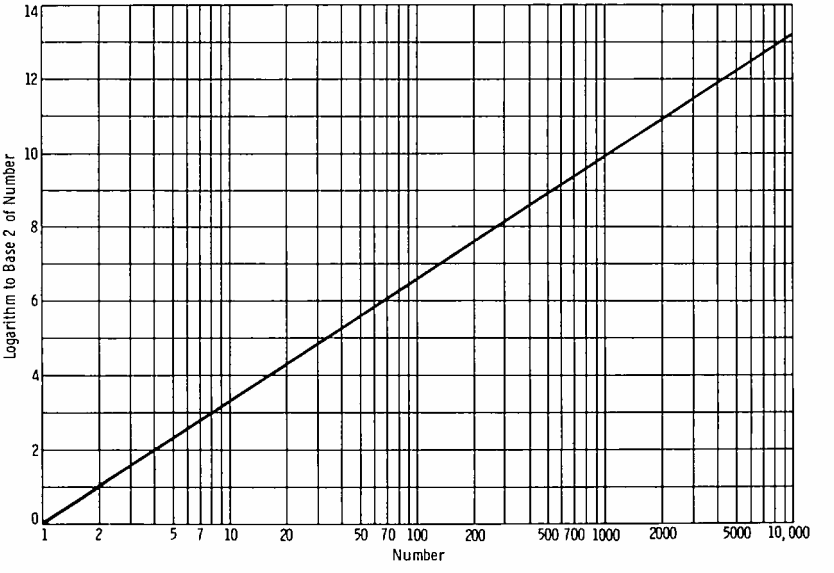
A11-8. When certain waveforms (such as clock pulses) must meet definite specifications and when timing information must be obtained.

A11-9. Actual quantizing error is *fixed* by the number of bits in a binary word. For example, an 8-bit binary word has a maximum quantizing error of 1/512, or 1 part in 512, at full level. However, this error is a *greater percent of the total* for a low-level signal than for a high-level signal.

A11-10. For an analog system, 20 IEEE units of color subcarrier superimposed on a ramp or stairstep is normally used. This should be increased to 40 IEEE units for a digitized video signal. Also, the phase of the superimposed color subcarrier should be made random.

APPENDIX C

Logarithms to Base 2



Index