A DEVICE FOR QUANTIZING, GROUPING, AND CODING AMPLITUDE-MODULATED PULSES

by

Leon G. Kraft, Jr.

B.S. in E.E., University of Pennsylvania
(1944)

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
(1949)

Signature of Author ___
Department of Electrical Engineering
May 20, 1949

Certified by ______
Thesis Supervisor

______
Chairman, Department Committee on Graduate Students

ACKNOWLEDGMENT

The author wishes to thank Prof. J. B. Wiesner for his guidance in supervising this thesis, and Mr. T. P. Cheatham, Dr. A. B. Macnee, and others of the Research Laboratory of Electronics of M.I.T. who contributed many helpful suggestions and ideas.

ABSTRACT

The theory of information transmission proposed by Tuller, Shannon, and Wiener (in 1948) indicated a need for coding of the original message signals into signals suitable for the transmission channel in use. The problem of finding the optimum coding for the purpose of storing the information or for transmitting it under certain transmission conditions is studied in this paper. A partially mathematical description of possible binary codes is developed. The unsolved problem of finding the optimum (shortest) code directly, once the probabilities of the message symbols are known, is discussed with a geometrical representation.

The interesting idea of taking longer parts of the messages and coding these, instead of coding each message symbol separately, is discussed briefly with some illustrations of the advantages to be expected.

A device for coding small groups of message symbols is described. The device consists of a cathode-ray tube, mask, and photo-cell, with associated circuits so arranged that the electron beam of the cathode-ray tube follows certain edges between opaque and transparent areas of the mask. The amplitude-modulated pulses fed as a signal to this device determine which portions of the mask are followed by the electron beam.

TABLE OF CONTENTS

Page

CHAPTER I - CODING

CHAPTER II - CODING IN GROUPS

CHAPTER III - A DEVICE FOR CODING GROUPS OF PULSES

CHAPTER I

CODING

Introduction

It has been suspected for some years that the original Hartley Law placing a limit on the amount of information which may be transmitted through a given bandwidth in a given time was not complete.[1*] The development of new forms of modulation (frequency modulation in 1935, and pulse modulation in 1939 and during the war) has led to much theoretical investigation on the subject of communication channel capacity. This original research has led to a revision of the Hartley Law and to new theories regarding the efficiency of communications systems.[1,2,3]

The new theories make an important point that was not stated clearly before. This point is the definition or development of exactly what is meant by "information." There is general agreement among the various authors[2,3,4] that information conveying processes are based on "choice" or "selection." That is, information is conveyed when a transmitter makes selections among a group of possible choices and a receiver is "informed" of these selections. For example, in writing, a sequence of choices from among

[*] Superscripts refer to references listed in the Bibliography.

the letters of the alphabet is made. The reader is informed of these choices through the medium of the written page and the mechanism of the eye and nerves. The process conveys information. A simpler piece of information is a yes or no answer to a question. Using the basic idea of choice or selection, which is derived from experience, Shannon[2] and Fano[4] have developed a mathematical functional relationship between the amount of information and the nature of the possible choices. Wiener,[3] by the expert application of measure theory, arrives at the same functional form. It is the development of this mathematical relation which has led to a clearer understanding of how much "information" a source is producing and how much information a given communications system can transmit.

The equation relating information and independent basic choices is:

$$H = -\sum_{i} p(i) \log_2 p(i) \qquad (1)$$

where H is the average amount of information per choice, and p(i) is the a priori probability of the i^{th} of the possible choices. In the special case where there are only two choices and the a priori probability of each is 0.5, H is equal to:

$$H = -\tfrac{1}{2} \log_2 \tfrac{1}{2} - \tfrac{1}{2} \log_2 \tfrac{1}{2} = \log_2 2 = 1.0$$

This amount of information, the amount conveyed by the selection of one of two equally likely possibilities, is usually taken as the unit of information and is called one "bit." As a matter of fact, Fano has derived Equation (1) by starting with the definition of one unit of information as a selection between two equally likely choices.[4]

Another property of this measure of information is that it is a maximum when all the $p(i)$ are equal. If there are N possible choices, all equally probable,

$$p(i) = \frac{1}{N}$$

and

$$H_{max} = - \sum_{i=1}^{N} \frac{1}{N} \log_2 \frac{1}{N} = \log_2 N.$$

This property is in accord with the intuitive judgment that events are most "uncertain" when all the possibilities are equally likely to occur. It is this property which leads to the deduction that present communications systems do not make the optimum use of the power, bandwidth, and time expended for the transmission of the ensemble of messages for which they are designed. For, while it is true that most communications systems are designed to transmit a given set of possible choices (regardless of the frequency of occurrence of each choice), there is no reason to assume that information sources produce the choices with equal probability.

Purpose of Coding

In order to design the communication system, it is first necessary to study the nature of the messages to be transmitted on a statistical basis. This allows the computation of the actual average rate of the information which it is desired to transmit. The capacity of the system then need be no greater than necessary for this rate. However, in order to operate at this maximum rate, the source feeding the channel must have the statistical structure which maximizes the information rate in the channel. If the source or sources to be used do not have this structure, it is necessary to introduce a transducer that will "match" the source to the channel. The The transducer here is somewhat analogous to a transformer which is designed to transfer maximum power from a generator to a load; in the present case, the source is matched to the channel so that maximum information is transferred.

The matching transducer is often called a "coder" because most generally its output is quite different from the input in form. For example, the human telegraph operator performs a coding of the alphabet into electrical impulses of a different nature. The human shorthand writer codes spoken sounds into symbols which may be written down more quickly than the alphabet code for the same sounds. In this case, the recording channel, arm, hand, pen, and paper, does not have enough capacity for the alphabet code, and a code better matched to the channel must be used. Still a third example of coding is that used by the Western Union Company in transmitting a large group of greeting and holiday messages by merely sending a code number. It might be mentioned that even fewer numbers, on the average, would be needed to transmit the same messages if the probabilities of the individual messages were found and the codes then arranged to match these probabilities to the channel.

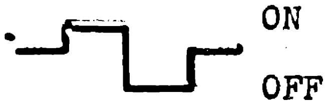
The purpose of coding, then, is to make the transmission system more efficient. As explained by Shannon,[2] the coder matches the source to the channel. In general, it is not possible to achieve an exact match (so that the channel is carrying information at its maximum possible rate), but it is possible to approach as near to an exact match as desired. The necessity and success of a coder depend, as outlined briefly above, on two factors: first, there must be mismatch between source and channel for any saving at all to be possible, and second, the saving in time, bandwidth, or power must be sufficient to warrant the increased complexity and cost of the coding transducer.

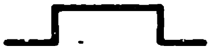
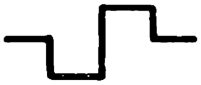
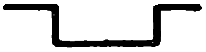
An Example of Coding (PCM)

The general concept of coding can be illustrated with a simple discussion of the coding characteristics of Pulse Code Modulation (PCM). PCM has been developed for telephone communications by the Bell Laboratories.[7,8] In this system, the amplitude of a signal voltage is sampled, and a group of seven voltage pulses is transmitted to represent each sample amplitude. The pulses transmitted are of the "on-off" type; i.e., there are just two possible amplitudes for each pulse. The groups of seven on-off pulses can be made to represent, therefore, exactly one-hundred and twenty-eight (2^7) different amplitude levels. The same length code (seven pulses) is devoted to each sample of amplitude.

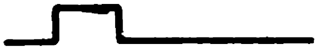
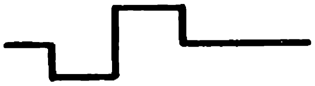
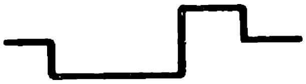
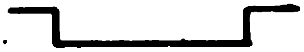
In a system handling a restricted class of messages, such as voice communications, it is conceivable that the one hundred twenty-eight amplitude levels would not occur with equal probability and that there would be some conditional probabilities between successive choices. (A recently published partial result substantiates this.)[5] If the coding were arranged so that the most probable amplitudes were represented by short codes, less time would be necessary to transmit messages on the average.

For example, suppose it is desired to transmit messages made up of just <u>four</u> amplitudes. In a straightforward binary system, the following on-off code pulses would be sent:

	p(i)	code
Amplitude #1	0.25	

(continued)	p(i)	code
Amplitude #2	0.25	
Amplitude #3	0.25	
Amplitude #4	0.25	

If the four amplitudes occurred independently and with equal probability, the above codes would be as short as possible and each would transmit two units of information. However, if the probabilities are not equal and independent, it may be possible to devise a better code. As an example, consider the case for the following probabilities and codes:

	p(i)	code
Amplitude #1	0.500	
Amplitude #2	0.250	
Amplitude #3	0.125	
Amplitude #4	0.125	

The amount of information being generated per selection of the original amplitudes is

$$H = -\sum_{1} p(i) \log_2 p(i) = 1.75 \text{ units of information per choice.}$$

If the original code were used, there would be two code digits for each choice or 0.875 units of information conveyed by each code digit. If a new code such as the one shown above were used, we find the <u>average</u> code length would be

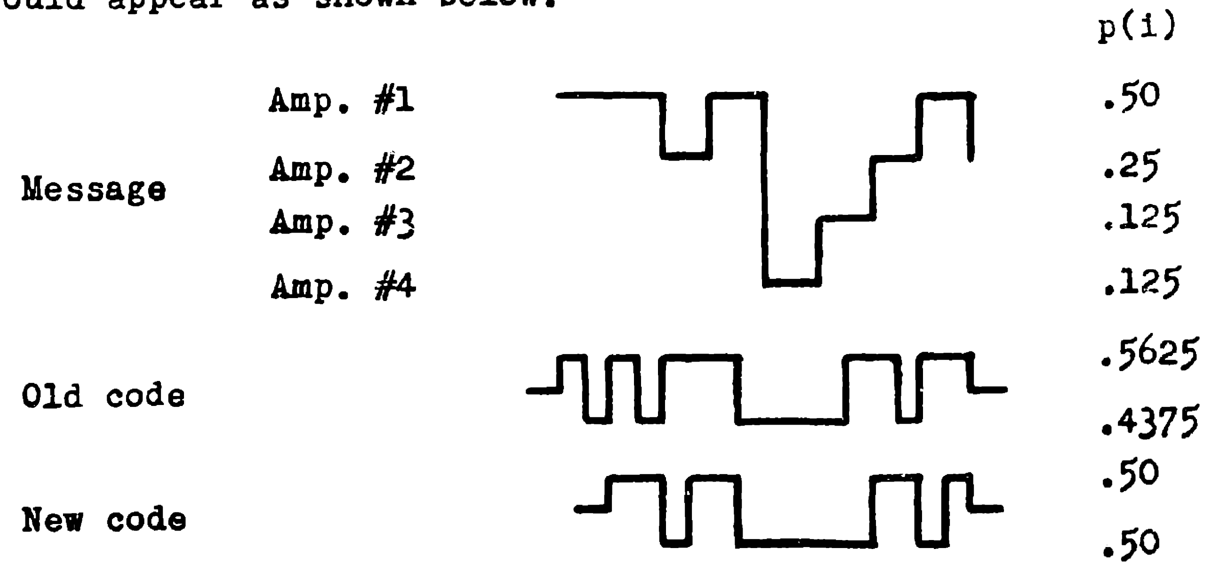
$$L_{ave} = \sum_{i} p(i)\ b(i)$$

where p(i) is the probability of each code, and b(i) is the number of digits in each code. For the code given, the average code length (L_{ave}) would be:

$$L_{ave} = \sum_{i} p(i)\ b(i) = 1.75 \text{ digits/code.}$$

Thus, in this case, we see that 1.00 unit of information is conveyed by each code digit. The code suggested is optimum, since (as has already been mentioned) the maximum information rate is $\log_2 N$, and where the signal transmitted has just two levels, as in the above codes, $\log_2 N = 1.00$.

The important point about the new code is that it is <u>shorter</u>, on the average, than the old. It is possible to transmit a given message in less time with the new code. For example, the message and the two codes discussed above would appear as shown below.



Binary Coding of Equally-Probable Choices

It has been shown by Fano[4] that the measure of information $H_N = \sum_{1} p(i) \log_2 p(i)$ is equal to the average number of equally likely binary selections required to specify a particular selection from a group of N possible choices. In other words, the average length of a binary code representing a particular one of N possible choices would just equal H_N -- if the coding were perfect.

The binary coding cannot be made perfect for all cases. In fact, perfect codings are special cases. Even in the restricted case where all the possibilities N are equally probable and independent, it is found that binary codes will not transmit one unit of information per code digit. As examples, the amount of information per code digit is calculated for the case of N = 3, 4, 5 and codes developed in the manner suggested by Fano.

Possibilities	p(i)	Code
1	1/3	1
2	1/3	01
3	1/3	00

$$H_N = \log_2 N = \log_2 3 = 1.585$$

$$L_{ave} = \sum_{1} p(i)\, b(i) = 1.667$$

$$\frac{H_N}{L_{ave}} = \text{ave. information per code digit} = 0.951$$

1	1/4	11
2	1/4	10
3	1/4	01
4	1/4	00

Possibilities	p(i)	Code

$$H_N = \log_2 N = \log_2 4 = 2.000$$

$$L_{ave} = \qquad 2.000$$

ave. information per code digit = 1.000

1	1/5	11
2	1/5	10
3	1/5	01
4	1/5	001
5	1/5	000

$$H_N = \log_2 N = \log_2 5 = 2.320$$

$$L_{ave} = \sum_i p(i)\, b(i) = 2.400$$

ave. information per code digit = .967

These results can be generalized for each of the N equally likely. The theoretical information in a choice from N possibilities is $\log_2 N$. The average length of the shortest binary code can be found as follows:

Let α = largest integer such that $2^{\alpha} \leq N$.

2^{α} of the N possibilities may be coded by codes of α digits.

The remaining possibilities $(N - 2^{\alpha})$ may be given codes identical with an equal number of the first group and then one digit added to each code which occurs twice to distinguish them.

Thus, $2(N - 2^{\alpha})$ codes are $(\alpha + 1)$ digits long, while $N - 2(N - 2^{\alpha})$ are still α digits long.

$$L_{ave} = \frac{(\alpha + 1)\, 2(N - 2^{\alpha}) + \alpha(N - 2N + 2^{\alpha})}{N}$$

$$= \frac{2N + \alpha N - 2^{\alpha + 1}}{N}$$

$$= 2 + \alpha - \frac{2^{\alpha + 1}}{N}$$

This code can be shown to have a minimum average length by noting that the code lengths must satisfy the expression:

$$\sum_{i = 1}^{N} \frac{1}{2^{b(i)}} \leq 1$$

If any of the b(i) are decreased, the above restriction will be violated unless one or two of the other codes are lengthened. Since the probabilities are equal, nothing is gained by shortening one code at the expense of the others. (The above expression will be derived later.)

The difference between the average information capacity of these codes, which is one unit of information per code digit, and the actual information which they represent is:

$$D = L_{ave} - H_N \quad \text{units of information (bits)}$$

$$D = 2 + \alpha - \frac{2^{\alpha + 1}}{N} - \log_2 N$$

For simplification, let

$$x = \alpha + 1 - \log_2 N$$

$$D = 1 + x - 2^x$$

In this expression, x varies between one and zero as N increases. Note that each time $x = 0$, α increases by one, so that it always satisfies the original definition of being the largest integer such that $2^{\alpha} \leqq N$.

Figure 1 is a plot of this expression. The maximum of this expression may be found by differentiation:

$$\frac{dD}{dx} = 1 - 2^{x} \ln 2$$

$$x_m = \log_2 \frac{1}{\ln 2} = 0.529$$

and

$$D_{max} = 1 + 0.529 - 2^{0.529} = 0.086 \text{ units of information}$$

This means that if the worst possible number of things is chosen to be coded, the best binary code may still require a channel capacity 0.086 bits per code greater than the actual information being carried by each code. If a great many codes are sent, a great amount of information capacity is lost over all. However, the percentage loss is rather small. That is, the number of bits of information lost per binary digit transmitted is small.

The maximum loss will occur for an infinite number of N_w, which may be found as follows:

$$x_m = 0.529 = \alpha + 1 - \log_2 N_w$$

$$\log_2 N_w = \alpha + 0.471$$

$$N_w = 2^{\alpha} \times 2^{0.471}$$

$$= 2^{\alpha} \times 1.386$$

A partial list of the worst choices of N is:

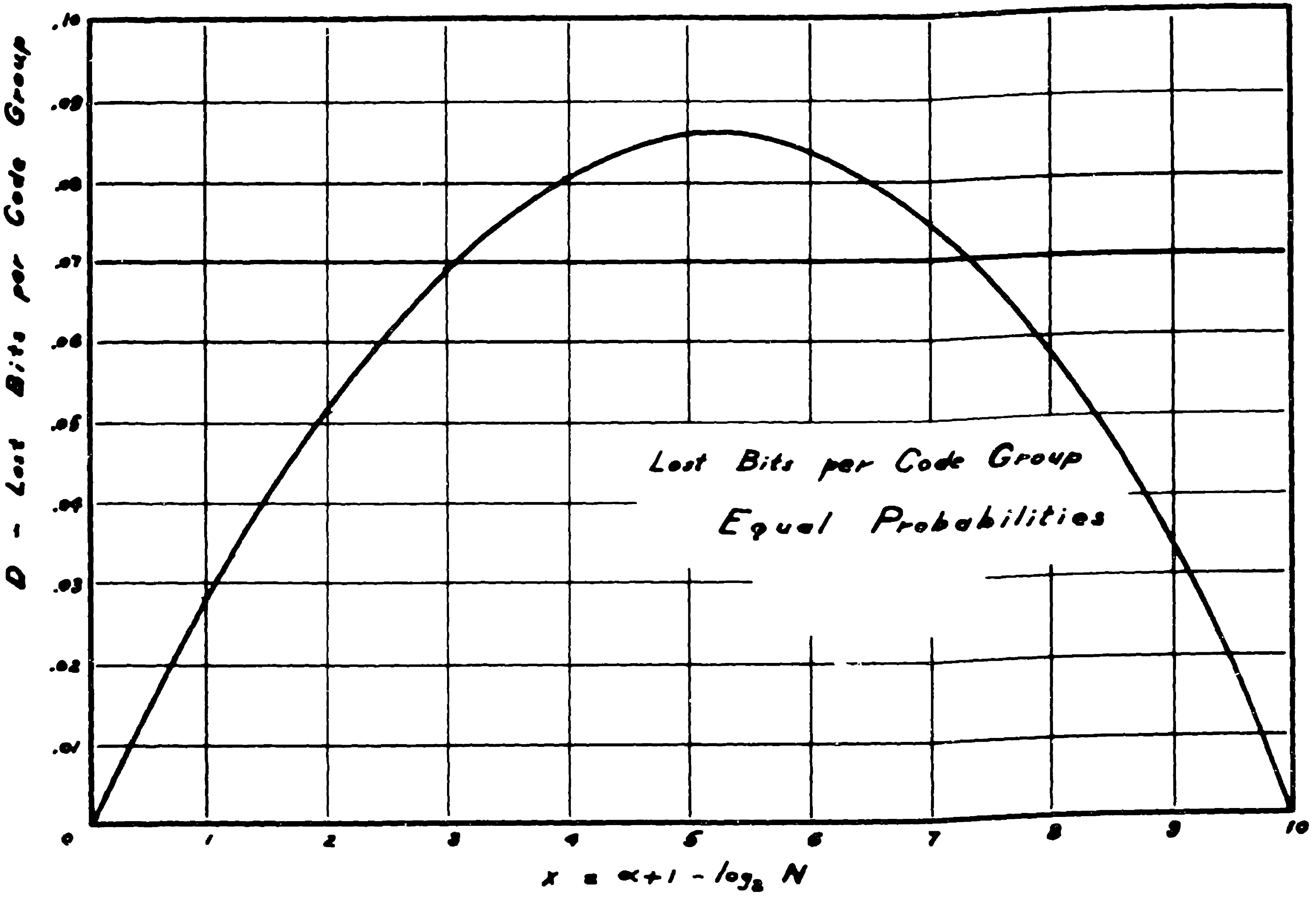

Lost Bits per Code Group
Equal Probabilities
D - Lost Bits per Code Group
.10
.09
.08
.07
.06
.05
.04
.03
.02
.01
0
1
2
3
4
5
6
7
8
9
10
$x = \alpha + 1 - \log_2 N$

Figure 1

α	N_w	N_w (nearest integer)
0	1.386	
1	2.77	3
2	5.54	5 or 6
3	11.1	11
4	22.2	22
5	44.3	44
6	88.7	89
7	177.2	177
8	355.	355
9	709.	709
10	1418.	1418

The average information capacity lost per code digit is, of course:

$$\frac{L_{ave} - H_N}{L_{ave}}$$

$$= 1 - \frac{H_N}{L_{ave}}$$

If the worst possible N is chosen, this is approximately equal to:

$$\frac{.086}{\alpha + 0.557} \qquad \frac{\text{lost bits}}{\text{binary code digit}}$$

For example, if $N = 3$, $p(i) = \frac{1}{3}$,

then

$$H_N = \log_2 3 = 1.585 \ \frac{\text{bits}}{\text{possibility}}$$

$$L_{ave} = 2 + \alpha - \frac{2^{\alpha + 1}}{N}$$

$$= 2 + 1 - \frac{4}{3} = 1.667 \ \frac{\text{digits}}{\text{code}}$$

$$D = L_{ave} - H_N$$

$$= 1.585 - 1.667 = .082 \quad \frac{\text{lost bits}}{\text{code}}$$

$$\frac{D}{L_{ave}} = \frac{.082}{1.667} = .049 \quad \frac{\text{lost bits}}{\text{code digit}}$$

If $N = 11$, $p(i) = \frac{1}{11}$

$$H_N = \log_2 11 = 3.460 \quad \frac{\text{bits}}{\text{choice}}$$

$$L_{ave} = 2 + 3 - \frac{16}{11} = 3.545 \quad \frac{\text{digits}}{\text{code}}$$

$$D = \qquad = .085 \quad \frac{\text{lost bits}}{\text{code}}$$

$$\frac{D}{L_{ave}} = \qquad = .024 \quad \frac{\text{lost bits}}{\text{code digit}}$$

On the other hand, the best possible values for N are those for which

$$x = 1 + \alpha - \log_2 N_b = 0 \quad \text{or} \quad 1 ,$$

$$\log_2 N_b = \alpha \quad \text{or} \quad \alpha + 1 , \quad \text{and}$$

N_b are the integral powers of 2.

The above development may be summarized as follows:

(1) The minimum length binary code for N independent, equally likely possibilities will involve extra information capacity unless N is an integral power of two.

(2) The worst possible N which may be chosen are: $N_w = 1.326 \times 2^{\alpha}$, where $\alpha = 1, 2, 3, \ldots.$

(3) The extra capacity is about 4.9% in the case of $N = 3$ and decreases roughly as $\frac{1}{\log_2 N}$. At $N = 89$, the extra capacity is only about 1.3%.

Present Coding Methods for Non-equal Probabilities

A much more important aspect of coding appears in the consideration of codes for possibilities which are not equally likely. This is important for two reasons: one, some partial results and present judgments indicate that equal probabilities are what occur in practice; two, if the same type codes as described above are used, more and more capacity is wasted as H_N decreases from its maximum where all the N probabilities are equal.

Two general approaches to this coding problem have been suggested.[2,4] The first is that of Shannon in which; the possible selections are arranged in order of decreasing probabilities, partial sums of the probabilities are formed, the sums expressed in binary notation, and enough digits of each binary number retained to form a unique code group. The following example illustrates this method:

Possibilities	p(i)	$P_s = \sum_{i=1}^{s-1} p(i)$	P_s (binary notation) and code
1	0.500	0.0000	.0
2	0.250	0.5000	.10
3	0.125	0.7500	.110
4	0.125	0.8750	.111

In the above, P_s is the sum of the p(i) up to but not including p(s). The code for each possibility is P_s, expressed in binary notation and limited to b(i) places where b(i) is the integer satisfying:

$$\log_2 \frac{1}{p(i)} \leq b(i) < 1 + \log_2 \frac{1}{p(i)}$$

This choosing of b(i) ensures that each code will differ from all the rest. However, as will be shown later. it does not ensure the shortest possible code.

The second method of coding, suggested by T. P. Cheatham of the communications group at the Research Laboratory of Electronics and developed by R. M. Fano,[4] proceeds as follows: the possibilities are arranged in order of decreasing probability, the group is split into two groups of as nearly equal probability as possible, the digits 0 and 1 are assigned to all members of each group as the first code digit, and this process is continued until each group contains only one member. Using the same probabilities as above:

Possibilities	p(i)	Code
1	0.500	0
2	0.250	10
3	0.125	110
4	0.125	111

The horizontal lines in the code show the successive divisions between groups of equal probability.

It will be noted that each method leads to the same code in this example. (This is the same example of perfect coding used earlier.) Moreover, the two methods have been shown, by Shannon and Fano respectively, to converge to the optimum coding as the number of possibilities is increased by using longer and longer sequences of the original message symbols. However, they do differ slightly, and it is considered worth noting that the second method may result in a shorter code under certain circumstances. Consider the following example:

p(i)	P_s	b(i)	Shannon Code	Fano Code
4/9	0	2	.00	0
2/9	4/9	3	.011	10
2/9	6/9	3	.101	110
1/9	8/9	4	.1100	111

It is at once seen that the Shannon code is <u>longer</u> than even a straightforward binary coding -- which corresponds to the first two digits of the given code. Further, the Fano code is still shorter than the simple binary coding. In simple binary coding, the average code length would be, of course, two digits. In the above case, the Fano code length would be:

$$L_{ave} = \sum_{i = 1}^{N} p(i)\ b(i)$$

$$= \frac{4}{9}(1) + \frac{2}{9}(2 + 3) + \frac{1}{9}(3)$$

$$= \frac{4}{9} + \frac{10}{9} + \frac{3}{9} = \frac{17}{9} = 1\frac{8}{9} \text{ digits.}$$

A short consideration will show that the b(i) selected in the Shannon method is too restrictive; that actually b(i) - 1 will suffice in many cases. For the equally probable case discussed above, the Shannon method would make the number of code digits equal to $\alpha + 1$ in all cases (α being the integer just less than $\log_2 N$). However, it has been shown that unique codes can be constructed in which $2^{\alpha + 1} - N$ codes are only α digits long.

Although the above discussion is intended to show the slight advantage of the Fano coding method over that used in Shannon's proof, it should not be construed as argument that the Fano code is optimum. As a matter of fact, the Fano coding method fails under certain conditions for very logical reasons. This failure may be proved by an example and the reason then seen more closely.

Suppose the following probability distribution exists:

p(i)	Fano Code	Better Codes	
0.4	11	1	11
0.1	10	011	101
0.1	011	0101	100
0.1	010	0100	011
0.1	001	001	010
0.1	0001	0001	001
0.1	0000	0000	000

The average length of the Fano code is:

$$4 - 1(.3) - 2(.5) = 2.7 \text{ digits}$$

The average length of the better codes is:

$$4 - 1(.2) - 3(.4) = 2.6 \text{ digits}$$

The reason the better codes exist is that, although the first digit of the Fano code separates the possibilities into two groups of probability exactly 0.5, this very operation forces the second digit to separate groups of widely different probabilities (part of the time). On the other hand, the better code makes an imperfect division by the first digit, but better divisions at least in the second digit. In other words, the better codes make better subdivisions into equally probable groups on the average.

The Problem of Finding the Shortest Code

The problem of finding the optimum (shortest) code has, unfortunately, not been solved. That is, there is no known direct method with which one may operate directly on the probabilities and be sure of obtaining the shortest code. However, the problem may be set up so that its nature is clear and a geometrical interpretation given. Moreover, this analysis leads to a suggestion for a modified coding method which seems to work.

Before trying to find the best code, it may be advantageous to determine the possible codes. The following analysis due to R. M. Redheffer leads to a geometrical interpretation of possible codes. It will be noted that the entire code may be specified by the number of individual codes of each length. For example, one entire code for

N = 4 is;

1	1
2	01
3	001
4	000

This may be specified by the set of numbers (1, 1, 2), which means one individual code of length one, one individual code of length two, and two individual codes of length three. Therefore, let the set of numbers $(n_1\ n_2\ n_3 \ldots n_k \ldots n_p)$ specify an entire code, where

n_k = No. of k-digit individual codes.

The problem of finding the possible entire codes is then the same as finding the <u>possible</u> sets of $(n_1\ n_2 \ldots n_k \ldots n_p)$. Since the codes consist of binary code groups, the problem is that of determining how many different binary code groups can be formed with k binary digits -- after the $n_1\ n_2 \ldots n_{k-1})$ have been chosen. Notice that all the individual codes are different, and that none starts with a series of binary digits exactly the same as an entire shorter code. That is, none of the longer individual codes contains any of the shorter ones as its beginning. Under these restrictions, it is apparent that there may be formed 2^k combinations of k binary digits less any combinations which start with a series of digits just like a shorter code. In other words,

$$n_k \leq 2^k - n_1 2^{k-1} - n_2 2^{k-2} - n_3 2^{k-3} \ldots n_{k-1} 2.$$

Suppose, for example, that $n_1 = n_2 = 0$, $n_3 = 3$; what value may n_4 have?

$$n_4 \leq 2^4 - 0 \times 2^3 - 0 \times 2^2 - 3 \times 2$$

$$n_4 \leq 16 - 6 = 10$$

There are 2 four-digit arrangements of the binary numbers which will have the same first three digits as one of the given three-digit codes. Since there are three of these already used three-digit combinations, 3 x 2 or 6 four-digit combinations are not allowable. A similar argument holds for each of the subtracted terms.

It should be noted that if the equality is satisfied in the above expression for n_k, then there are no longer codes which will not contain a shorter one. In other words, for this condition, $n_{k+1} = n_{k+2} = \ldots = 0$. This may be called an "exhausted" code. Mathematically:

$$n_p = 2^p - n_1 2^{p-1} - n_2 2^{p-2} - \ldots n_{p-1} 2.$$

The above expressions may be expressed more compactly by rearrangement:

$$\sum_{j=1}^{K} \frac{n_j}{2^j} < 1 \qquad (K < p)$$

$$\sum_{j=1}^{p} \frac{n_j}{2^j} = 1 \quad \text{(for an exhausted code)}$$

This form leads to the following geometrical interpretation. The second summation may be thought of as the scalar product of two vectors in p-dimensional space.

$$\overline{N} = n_1 \, \overline{x}_1 + n_2 \, \overline{x}_2 + \ldots n_p \, \overline{x}_p$$

$$\overline{A} = \frac{1}{2} \, \overline{x}_1 + \frac{1}{2^2} \, \overline{x}_2 + \ldots \frac{1}{2^p} \, \overline{x}_p$$

where $\overline{x}_1$, $\overline{x}_2$, ... $\overline{x}_p$ are unit vectors along the coordinate axes of the p-dimensional space.

$$\overline{N} \cdot \overline{A} = \sum_{j=1}^{p} \frac{n_j}{2^j} = 1$$

$$|N||A| \cos \Theta_{NA} = 1$$

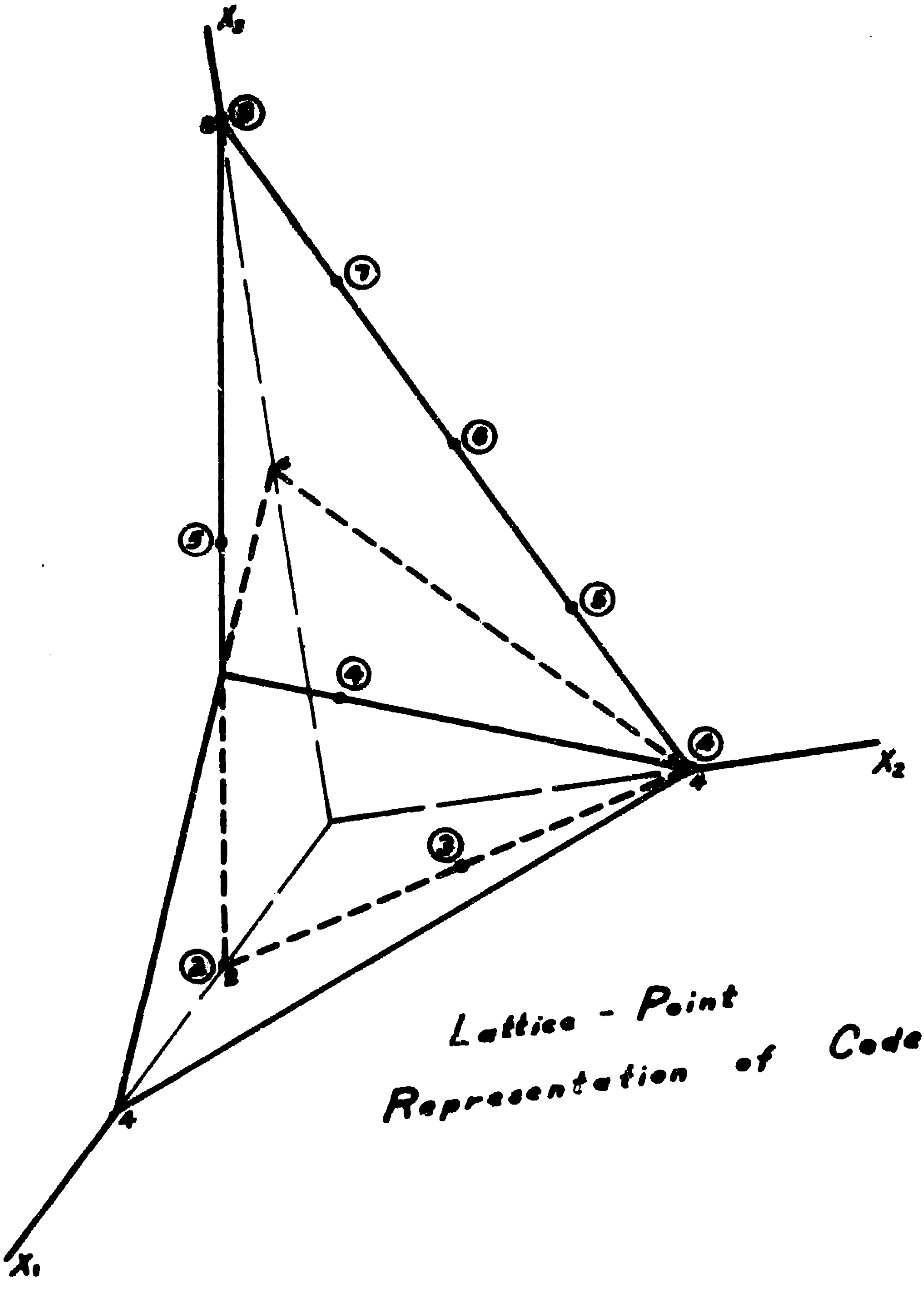
$$|N| \cos \Theta_{NA} = \frac{1}{|A|}$$

The locus of the ends of all N-vectors satisfying the last expression is, of course, a p-dimensional plane perpendicular to the A-vector at a distance of $\frac{1}{|A|}$ from the origin. Thus, the lattice-points (points whose coordinates are integers) on this plane represent possible exhausted codes. In a like manner, lattice points in the volume defined by this plane and the positive (all the n's are positive) extensions of the principal planes represent possible non-exhausted codes.

Figure 2 depicts the geometry in question for the case of p = 3. The plane with intercepts (2, 4, 8) is the surface with lattice-points, satisfying:

$$\sum_{j=1}^{p} \frac{n_j}{2^j} = 1 \, .$$

It is seen that there are six exhausted codes having p = 3 (maximum codes are three digits long). It is also noted that there are lattice points in the x_1x_2 plane for which



Lattice - Point Representation of Codes

Figure 2

$n_3 = 0$. These are exhausted codes of two digits maximum length. There is one code $n_1 = 2$ which is the only exhausted code of one digit maximum length. Thus, the figure for $p = 3$ includes as special cases all the possible lattice-points which represent codes of one, two, or three digits maximum length.

The question of how many sets of $(n_1\ n_2 \ldots n_p)$ are possible for a given N of possibilities may now be answered. The set of $(n_1\ n_2 \ldots n_p)$ must sum to N to satisfy this condition. That is, the total number of codes equals N.

$$\sum_{j=1}^{p} n_j = N$$

Geometrically, this condition is represented by a p-dimensional plane with intercepts on all the axes at N. Now the lattice points satisfying both conditions are those lying on the intersection of the two planes.

As an example, the plane for $N = 4$ is plotted on Figure 2, and its intersection with the previously described plane contains two lattice-points which are labeled ④. There are therefore two codes for four possibilities. One has a maximum of two digits, the other three digits.

Returning to the problem of the optimum code, it is found that the present p-dimensional picture does not lend itself to the representation of the set of probabilities which it is desired to code. That is, for this problem, it is more practical to choose an N-dimensional space in which the set of probabilities associated with the N possibilities which are being coded may be represented by the vector;

$$\bar{P} = p_1 \bar{x}_1 + p_2 \bar{x}_2 + \dots p_n \bar{x}_n .$$

A condition which must be satisfied by any set of (p_1 p_2 ... p_n) is that;

$$\sum_{i=1}^{N} p(i) = 1 .$$

All possible sets of probabilities are geometrically represented by a plane with intercepts on all principal axes equal to 1.

The same N-dimensional space may be used to represent the set of code lengths (b_1 b_2 ... b_r ... b_n), where b_r is the number of binary digits in the r^{th} individual code. Each individual code length must satisfy a condition equivalent to that derived by Redheffer in order that it be possible to write a unique binary number for the code. This condition has been derived as;

$$\sum_{j=1}^{n} \frac{n_j}{2^j} \leq 1 ,$$

and the equivalent condition in terms of the set of code lengths is;

$$\sum_{i=1}^{r} \frac{1}{2^{b(i)}} \leq 1 .$$

When the equality holds, an exhausted code has been reached.

Expressed in this way, there is no upper limit on the magnitude of b_r. However, if the whole code is for a given number (N) of possibilities, there is no point in making b_r excessively long. In other words, it is desirable to have an exhausted code. This condition on the b(i) may be derived as follows:

Let (b_1 b_2 ... b_{r-1}) be code lengths already chosen. Then, the number of new codes of length b_r that are possible is;

$$2^{b(r)} - \frac{2^{b(r)}}{2^{b_1}} - \frac{2^{b(r)}}{2^{b_2}} - \dots \frac{2^{b(r)}}{2^{b_{r-1}}}$$

The number of possibilities remaining to be coded is;

$$N - (r - 1)$$

and b_r must be chosen small enough so that the number of possible new codes does not exceed the number of things to be coded. That is;

$$2^{b(r)} \left[1 - \sum_{i=1}^{r-1} \frac{1}{2^{b(i)}} \right] \leq N - (r - 1)$$

Or, rearranging,

$$\sum_{i=1}^{r} \frac{1}{2^{b(i)}} + \frac{N - r}{2^{b(r)}} \geq 1.$$

The two conditions on the b(i) are then; first, the b(i) must be large enough to make the coding possible, and second, the b(i) should not be larger than necessary to result in an exhausted code. These two conditions mathematically reduce to the same expression only if r = N. That is,

$$\sum_{i=1}^{N} \frac{1}{2^{b(i)}} = 1 .$$

To find the average code length, given a set of probabilities $(p_1\ p_2 \ldots p_n)$ and a code $(b_1\ b_2 \ldots b_n)$, it is necessary to multiply corresponding p's and b's and sum. Thus, the average code length L_{ave} is;

$$L_{ave} = \sum_{i=1}^{N} p(i)\ b(i)$$

Geometrically, the set of p's $(p_1\ p_2 \ldots p_n)$ is represented by

$$\overline{P} = p_1\, \overline{x}_1 + p_2\, \overline{x}_2 + \ldots p_n\, \overline{x}_n ,$$

and the set of b's $(b_1\ b_2 \ldots b_n)$ by

$$\overline{B} = b_1\, \overline{x}_1 + b_2\, \overline{x}_2 + \ldots b_n\, \overline{x}_n ,$$

and the average code length by

$$\overline{P} \cdot \overline{B} = L_{ave} = \sum_{i=1}^{N} p(i)\ b(i) .$$

The restrictions discussed above are interpreted geometrically to mean that the tip of $\overline{P}$ must lie on the plane intersecting all the axes at 1. The restriction

$$\sum_{i=1}^{N} \frac{1}{2^{b(i)}} = 1$$

means that the tip of $\overline{B}$ must lie on a curved surface. This surface (a curve in two dimensions) is shown in Figure 3 for two dimensions and in Figure 4 for three dimensions.

The equation $\overline{P} \cdot \overline{B} = L_{ave}$ means that all the $\overline{B}$ points on a plane perpendicular to $\overline{P}$ represent codes which will have the same average length. That is,

$$\overline{P} \cdot \overline{B} = L_{ave}$$

$$|P|\ |B| \cos \theta_{PB} = L_{ave} \ .$$

The locus of all $\overline{B}$ which make $|B| \cos \theta_{PB}$ equal a constant (d) is the plane perpendicular to $\overline{P}$ at a distance d from the origin.

$$|B| \cos \theta_{PB} = d$$

and

$$|P|\ d = L_{ave}$$

The $\overline{B}$ points (these all lie on a particular curved surface), which lie in a plane perpendicular to the vector $\overline{P}$, will have equal code lengths which are proportional to the distance of the plane from the origin.

The problem of finding the optimum code is now to move the plane perpendicular to $\overline{P}$ out from the origin until it contains a lattice-point which is also on the

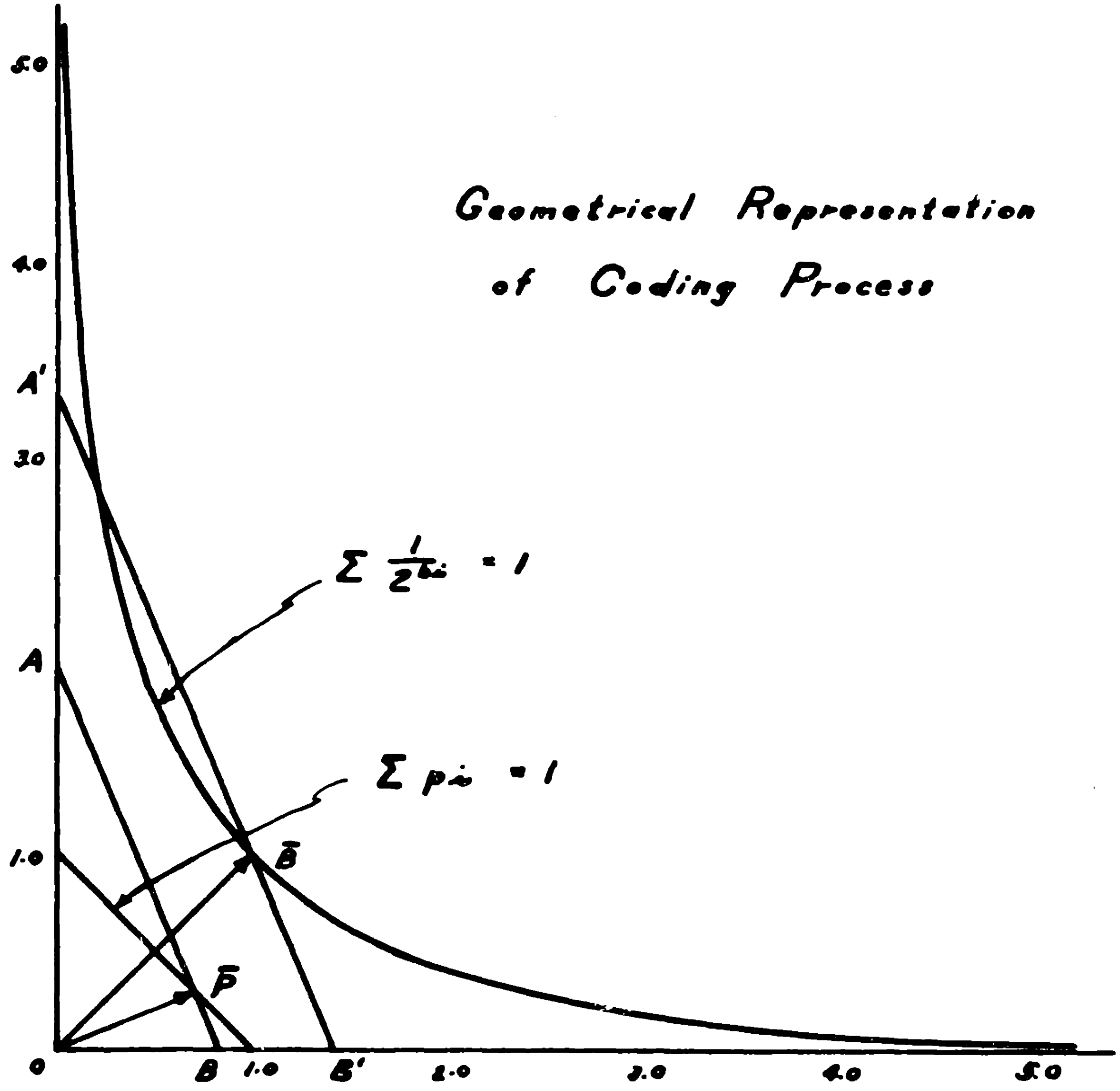
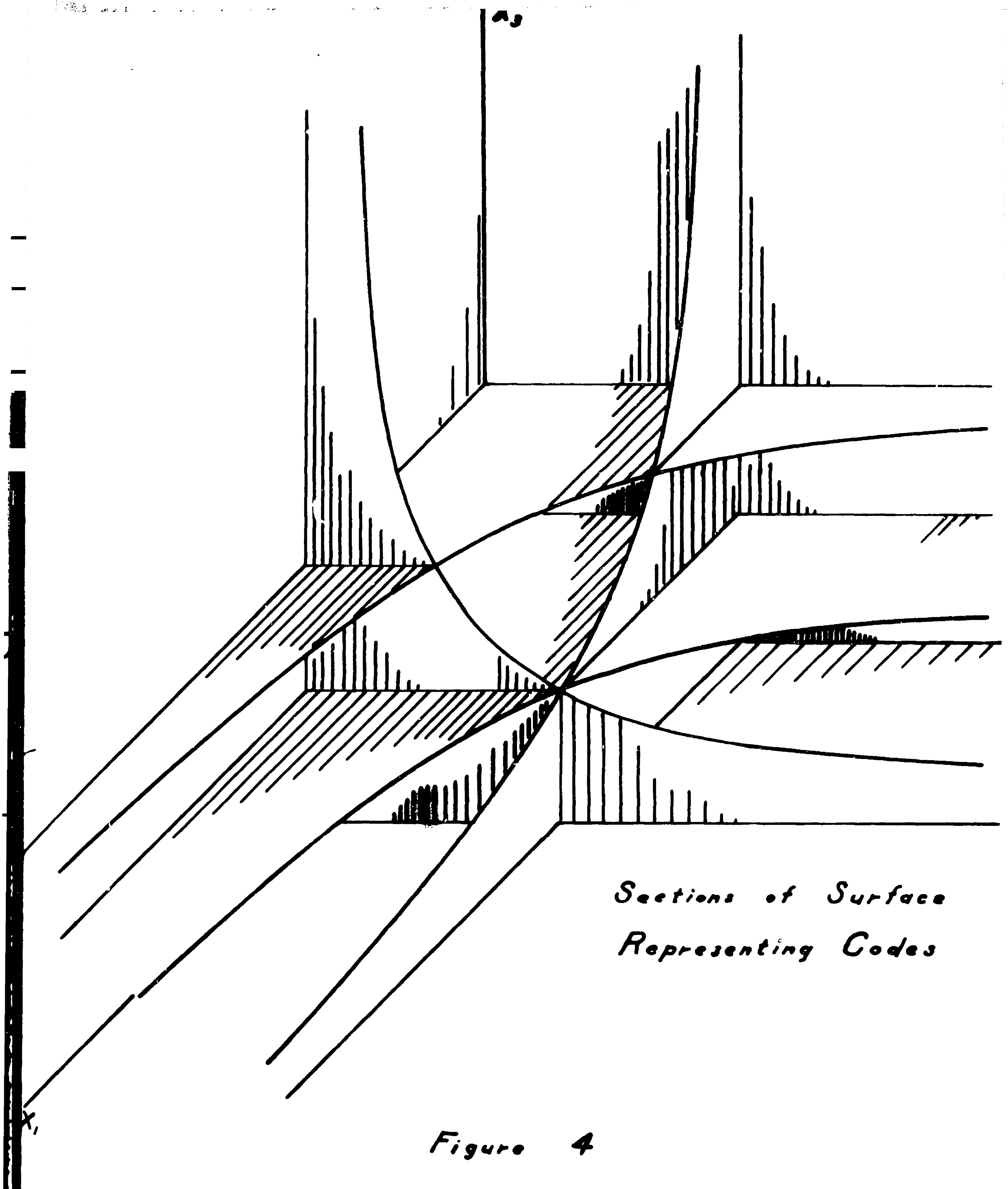

Geometrical Representation of Coding Process
5.0
4.0
A'
3.0
A
1.0
$\Sigma \frac{1}{2^{l_i}} = 1$
$\Sigma p_i = 1$
$\bar{B}$
$\bar{P}$
0
B
1.0
B'
2.0
3.0
4.0
5.0

Figure 3



Sections of Surface Representing Codes

Figure 4

surface of possible $\overline{B}$. In Figure 3, representing the two-dimensional case, the line AB is moved out from the origin to the position A'B'. In the new position, it contains the lattice-point (1, 1).

The moving line (or plane) contains none of the possible $\overline{B}$ points until it has moved out into a position tangent to the $\overline{B}$ surface. The coordinates of this point of tangency represent the minimum possible code lengths if the lengths are not restricted to integer values. (If the coordinates of the point of tangency happen to be integers, the rare case of a perfect code has occurred.) In other words, if the $(b_1\ b_2\ \ldots\ b_n)$ are allowed to change as continuous variables, the coordinates of the point of tangency of the plane perpendicular to $\overline{P}$ represent an absolute minimum length code. It is interesting to derive the coordinates of this point, since the result offers a check on the above representation of the coding process and also in some measure serves as a guide to a new coding procedure.

The problem of finding this minimum code length allowing the $(b_1\ b_2\ \ldots\ b_n)$ to vary continuously may be handled by ordinary minimization methods. The method of LaGrangian multipliers gives:

$$F\ (b_1\ b_2 \cdots b_n) = \sum_{i=1}^{N} p(i)\ b(i) - \lambda \sum_{i=1}^{N} \frac{1}{2^{b(i)}} + \lambda$$

to be minimized. There are then N partial derivatives equated to zero and the constraining equation,

$$\sum_{i=1}^{N} \frac{1}{2^{b(i)}} = 1 \ ,$$

to solve for the N values of b(i) and for λ.

$$\frac{\delta F}{\delta b(i)} = p(i) - \lambda \frac{\ln 2}{2^{b(i)}} = 0$$

$$\sum_{i=1}^{N} \frac{\delta F}{\delta b(i)} = \sum_{i=1}^{N} p(i) - \lambda \ln 2 \sum_{i=1}^{N} \frac{1}{2^{b(i)}} = 0$$

$$1 - \lambda \ln 2 = 0$$

$$\lambda = \frac{1}{\ln 2}$$

$$p(i) = \frac{1}{2^{b(i)}}$$

$$b(i) = \log_2 \frac{1}{p(i)}$$

This shows the point of tangency of the moving plane discussed above is the point with coordinates $(\log_2 \frac{1}{p_1}, \log_2 \frac{1}{p_2} \ldots \log_2 \frac{1}{p_n})$. The absolute minimum average code length , allowing non-integer values of code lengths is then:

$$L_{ave} = \sum_{i=1}^{N} p(i) \log_2 \frac{1}{p(i)}$$

or

$$L_{ave} = H_N \ .$$

H_N is the known result to this problem, and the above derivation merely enhances the validity of the equations from which it was derived. However, the geometrical interpretation also suggests a coding method which may be somewhat better than those previously tried.

Before discussing the modified coding method, it is interesting to note the position occupied by Shannon's code in the geometrical representation described. Shannon suggests that b(i) should be the integer that satisfies

$$\log_2 \frac{1}{p(i)} \leqq b(i) < 1 + \log_2 \frac{1}{p(i)}$$

This means selecting code lengths longer than the absolute optimum for most of the codes. In the geometrical sense, consider the N-dimensional cube formed by the lattice-points immediately surrounding the point $(\log_2 \frac{1}{p_1}$, $\log_2 \frac{1}{p_2} \ldots \log_2 \frac{1}{p_n})$. Shannon's point is the vertex of the cube farthest from the origin. The loss in choosing Shannon's code corresponds to the distance the moving plane is moved from the point of tangency to the farthest vertex. This distance is seen to be less than $\sqrt{N}$, and if the average code length is long (it varies as $\sqrt{N} \log_2 N$), the loss is relatively small. However, for a small N, the loss may be a considerable percentage of the whole code length.

A Suggested Coding Procedure

The geometrical representation shows the point representing the optimum code to lie in the volume contained by the $\overline{B}$ surface and the plane perpendicular to $\overline{P}$ through Shannon's point. This results in relatively few lattice-points to consider (depending on N and on the ratios of the given p(i), yet the problem of determining the best one seems to have no exact mathematical solution. The geometrical picture suggests that the lattice-points adjacent to the tangent point be considered as the first approximation to the best code. Shannon's point is, of course, one of these; however, any of the other adjacent lattice-points, if they satisfy

$$\sum_{i=1}^{N} \frac{1}{2^{b(i)}} \leqq 1 ,$$

will give better codes. This method is simple to apply -- all the code lengths may be chosen to satisfy;

$$\log_2 \frac{1}{p(i)} - 1 < b(i) \leqq \log_2 \frac{1}{p(i)}$$

up to a certain number, after which the condition

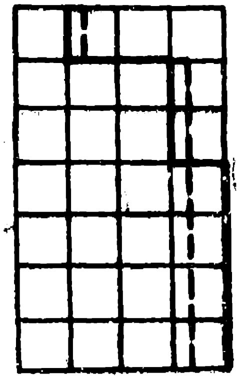
$$\sum_{i=1}^{N} \frac{1}{2^{b(i)}} \leqq 1$$

cannot be satisfied unless the b(i) satisfy Shannon's condition

$$\log_2 \frac{1}{p(i)} \leqq b(i) < 1 + \log_2 \frac{1}{p(i)} .$$

The coding method above described is easy to apply and involves little calculation and time. It will work, for example, in the example described earlier for which neither the Fano or Shannon methods gave the optimum code. The diagram below shows the steps in constructing the code according to the new method.

$p(i)$	$\log_2 \frac{1}{p(i)}$
.4	1.32
.1	3.32
.1	3.32
.1	3.32
.1	3.32
.1	3.32
.1	3.32



The dotted line shows the absolute optimum code lengths. The solid line shows the attempt to choose the integer just smaller than this. The solid line is forced out to the integer just greater than the optimum length at the fourth code.

It is noted that in certain instances the above procedure is not complete. First, if $p_1 > \frac{1}{2}$, the first code length must be one. Secondly, the last two code lengths must always be the same for an exhausted code, regardless of the relative values of p_n and p_{n-1}. Thirdly, for convenience, the probabilities are arranged in monotonically decreasing order.

For more refined work, the code lengths associated with all the lattice-points in the restricted volume found above must be calculated and compared. In this respect, limits may be placed on the lengths of the individual code lengths. These limits are the maximum and minimum values found by using the LaGrangian method as follows:

To find the maximum and minimum values of a particular b(r), notice that

$$\sum_{i=1}^{N} \frac{1}{2^{b(i)}} = 1,$$

expresses b(r) as an implicit function of the other b(i), and that

$$\frac{\partial b(r)}{\partial b(i)} = -\frac{2^{b(r)}}{2^{b(i)}} \qquad i = 1, 2 \ldots N \text{ (except } r).$$

Maximize (or minimize) this function, b(r) = F b(i), subject to

$$\frac{1}{|P|} \sum_{i=1}^{N} p(i) \left[b(i) - b(s_1)\right] = 0 \;,$$

which is the condition that the maximum and minimum of each b(i) are on the plane through Shannon's point. $b(s_1)$ are the coordinates of Shannon's point. Thus,

$$-\frac{2^{b(r)}}{2^{b(i)}} - \lambda \frac{p(i)}{|P|} = 0 \qquad i = 1, 2 \ldots N \text{ (except } r).$$

Summing,

$$-2^{b(r)} \left[1 - \frac{1}{2^{b(r)}}\right] - \frac{\lambda}{|P|} \left[1 - p(r)\right] = 0 \;,$$

and

$$-\frac{\lambda}{|P|} = \frac{2^{b(r)} - 1}{1 - p(r)} \;.$$

Substituting the values of the b(i) given by the partial derivative equations in the conditional equation gives:

$$\frac{p(r)}{|P|}\left[b(r) - b(s_r)\right] +$$

$$+\frac{1}{|P|}\sum_{\substack{i = 1 \\ \text{except } r}}^{N} p(i)\left[b(r) + \log_2 \frac{1 - p(r)}{2^{b(r)} - 1} + \log_2 \frac{1}{p(i)} - b(s_i)\right] = 0$$

$$b(r) - \sum_{i = 1}^{N} p(i)\, b(s_i) + \left[1 - p(r)\right] \log_2 \frac{1 - p(r)}{2^{b(r)} - 1} +$$

$$+ H_N - p(r) \log_2 \frac{1}{p(r)} = 0$$

$$b(r) + \left[1 - p(r)\right] \log_2 \frac{1 - p(r)}{2^{b(r)} - 1} = \sum_{i = 1}^{N} p(i)\, b(s_i) - H_N +$$

$$+ p(r) \log_2 \frac{1}{p(r)}$$

This is a transcendental equation in b(r) and has two roots, one of which is the maximum b(r), the other the minimum. Note that if a lattice-point representing a code better than Shannon's has been found (by any method whatsoever), its coordinates may be substituted in place of the $b(s_i)$ above and the range of b(r) reduced accordingly.

CHAPTER II

CODING IN GROUPS

Introduction

The first chapter has outlined the problem of coding N possibilities into N discrete binary codes. This has been done without thought of operating on the possibilities in any way and without consideration of what "source" is choosing the possibilities, It is the purpose of this chapter to consider the result of one form of operation upon the possibilities before coding.

The nature of this operation may be called "grouping." This means that, given a sequence consisting of choices from the N possible choices, groups of these choices are treated as units in the coding. This method of operating on the sequence of choices was suggested by T. P. Cheatham of the Research Laboratory of Electronics. The following is an example of this scheme:

Choices	P_i	Binary Code
1	0.333	1
2	0.333	01
3	0.333	00

L_{ave} = 1.667 binary digits

Groups of Choices	P_g	Binary Code
11	0.111	111
12	0.111	110
13	0.111	101
21	0.111	100
22	0.111	011
23	0.111	010
31	0.111	001
32	0.111	0001
33	0.111	0000

$$L_{ave} = 1.611 \ \frac{\text{binary digits}}{\text{original choice}}$$

It is seen that grouping the choices before coding reduces the code length (in the above example) and may be a method of reducing the needed capacity of a transmission system.

The improvement in code length shown by the example above is due to the nature of the numbers selected. A binary code can be made better for nine things than for three because of the relations among the powers of two and the powers of three. There is, in group coding, however, a more fundamental improvement also to be expected. This is the approach of the information rate, as calculated for a group of choices, to the actual information rate of the source. This has been thoroughly discussed by Shannon, who has proved that

$$H_g = \frac{1}{g} \sum_g P_g \log_2 \frac{1}{P_g} ,$$

the information rate calculated for groups of g choices or symbols, is a monotonic decreasing function of g. Therefore, if groups of choices are coded efficiently, it may be possible to reduce the average code length to H_g.

Grouping of Equally-Likely Independent Choices

The grouping of choices increases the number of possibilities to N^g, where g is the number of choices in the group. If the substitution $N = N^g$ is made in the equations developed for calculating the average code length for equally probable possibilities in Chapter I, and the result divided by g, the formulas hold for the present case. That is, the average code length for the N^g is;

$$L_{ave} = \frac{2 + \alpha - \frac{2^{\alpha+1}}{N^g}}{g} \quad \text{binary digits.}$$

The difference between this length and the theoretical minimum is;

$$D = \frac{2 + \alpha - \frac{2^{\alpha+1}}{N^g} - g \log_2 N}{g} \quad \frac{\text{binary digits}}{\text{original choice}} .$$

Letting

$$x' = \alpha + 1 - g \log_2 N ,$$

$$D = \frac{1 + x' - 2^{x'}}{g}$$

In these expressions, α is the integer just less than $\log_2 N^g$.

The loss in the code is seen to be similar to the loss without grouping, except for the g in the denominator. It is seen that increasing the number of choices in a group reduces the loss in coding approximately as $\frac{1}{g}$. The maximum and minimum values of this function may be found as functions of N or g. The equation may be normalized and plotted, in intervals of constant α , as

$$g\ D = 1 + x' - 2^{x'} ,$$

which is the same as Figure 1 except that g appears in the expression for x'. Proper use of this curve enables computation of the results of grouping choices that are equally probable. An example of this computation has been plotted in Figure 5 for N = 3. The dotted curve shows the theoretical limit (if g varies continuously). The solid lines connect the actual results which are obtained for integer values of g.

It is interesting, and important, to notice that the curve of actual values is *not* monotonically decreasing. In other words, although grouping of choices tends to improve the coding as the groups are made larger, the nature of binary (or other digital) coding may result in a worse code if the group's size is not increased sufficiently. In the case of N = 3 equally-probable possibilities, the group of five choices may be coded more efficiently than groups of six, seven, eight, or nine choices. This fact is important, because there would be

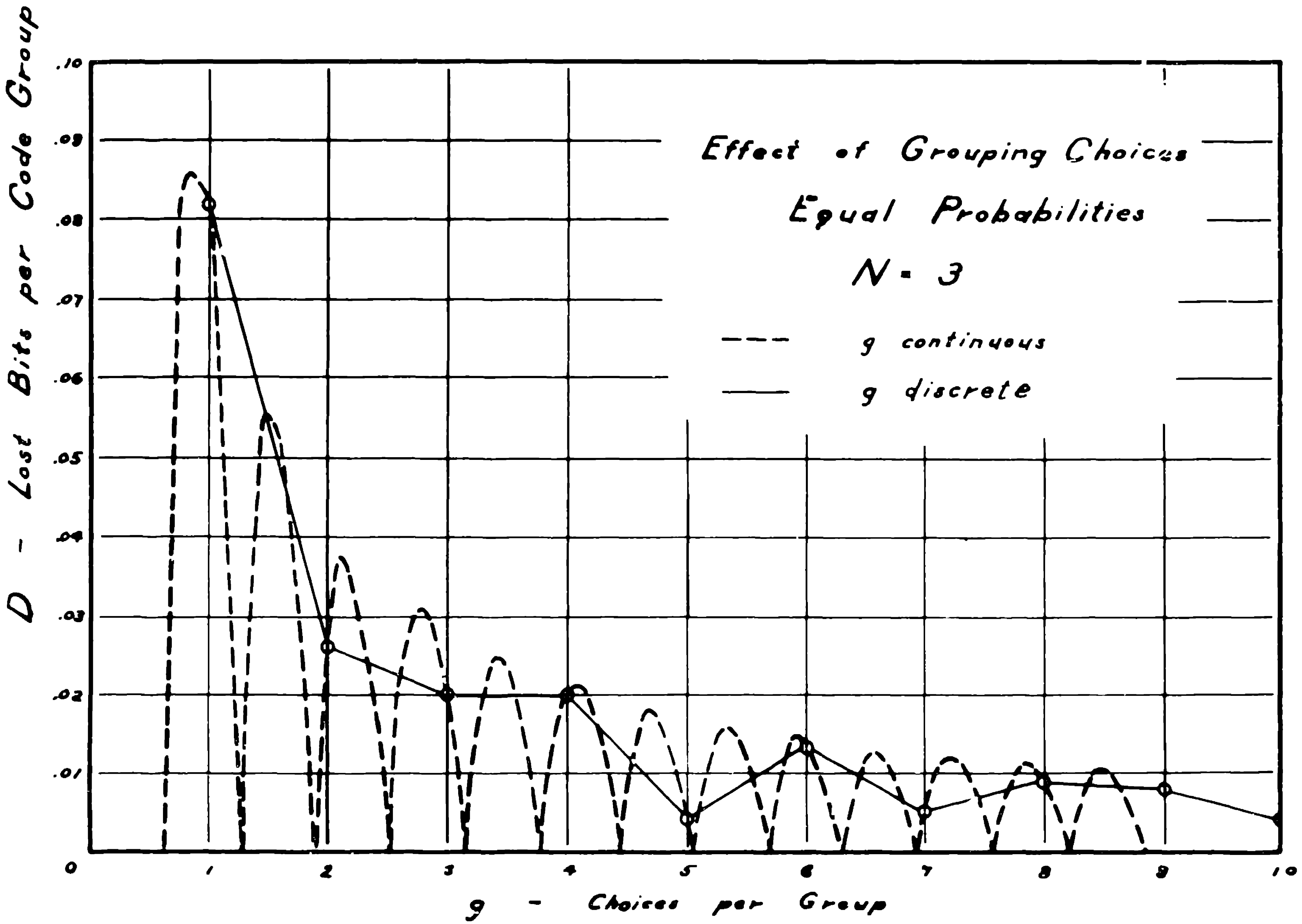

Effect of Grouping Choices
Equal Probabilities
N = 3
g continuous
g discrete
D - Lost Bits per Code Group
.10
.09
.08
.07
.06
.05
.04
.03
.02
.01
0
1
2
3
4
5
6
7
8
9
10
g - Choices per Group

Figure 5

a great deal wasted in building equipment, for example, to group and code beyond g = 5 for this special case.

Two Simple Examples of Grouping Choices with Unequal Probabilities

It is clear that grouping in the case of non-equally-likely choices is a difficult operation to evaluate, since it is difficult to find the optimum codes in this case. Nevertheless, Shannon has proved that the error in coding is less than $\frac{1}{g}$ plus the difference between the information rate calculated for groups of length g and the true rate. In order to show the results of grouping with unequal probabilities, two cases have been partially calculated. The results are shown in Figures 6 and 7.

Figure 6 is plotted for the case of three choices with probabilities $p_1 = 3/6$, $p_2 = 2/6$, and $p_3 = 1/6$. D is the number of extra binary digits needed per original choice. It is seen that grouping reduces this loss. Figure 7 is a plot of the same quantities for the original probabilities $p_1 = 7/10$, $p_2 = 2/10$, $p_3 = 1/10$. Here, the curve is not monotonically decreasing and indicates the same type oscillatory behavior as the curve for $p_1 = p_2 = p_3 = 1/3$. It is noted in this case that the original code is about 12% long, while the code after grouping two choices is only about 0.8% long.

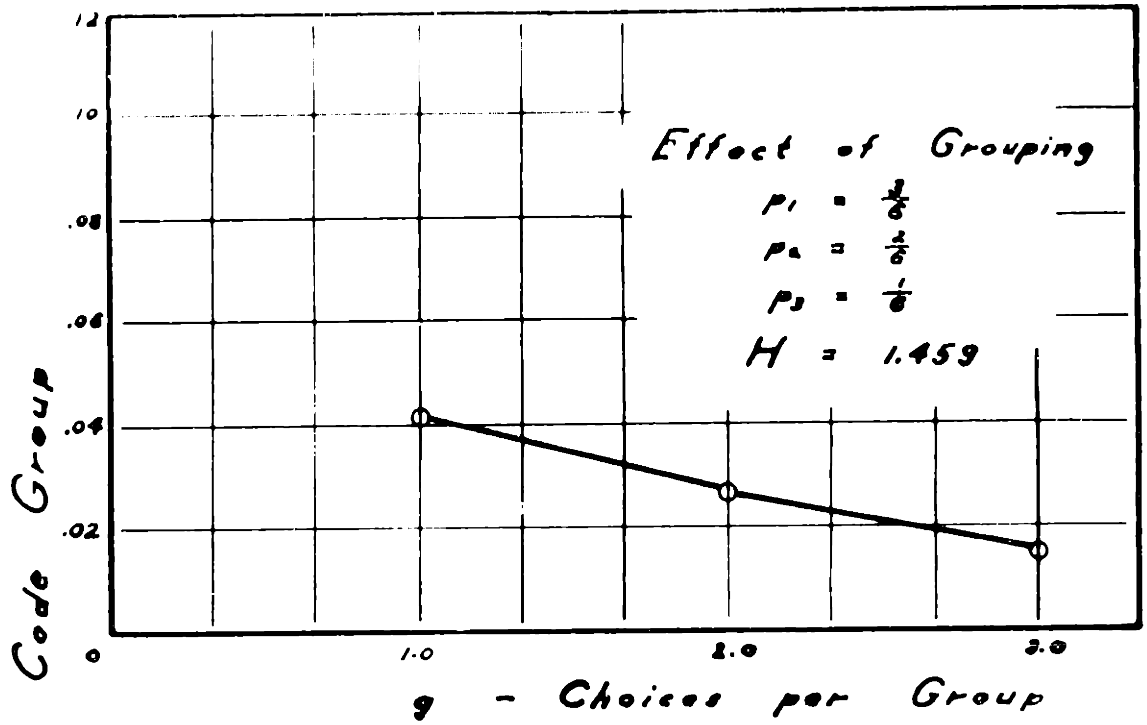


Figure 6

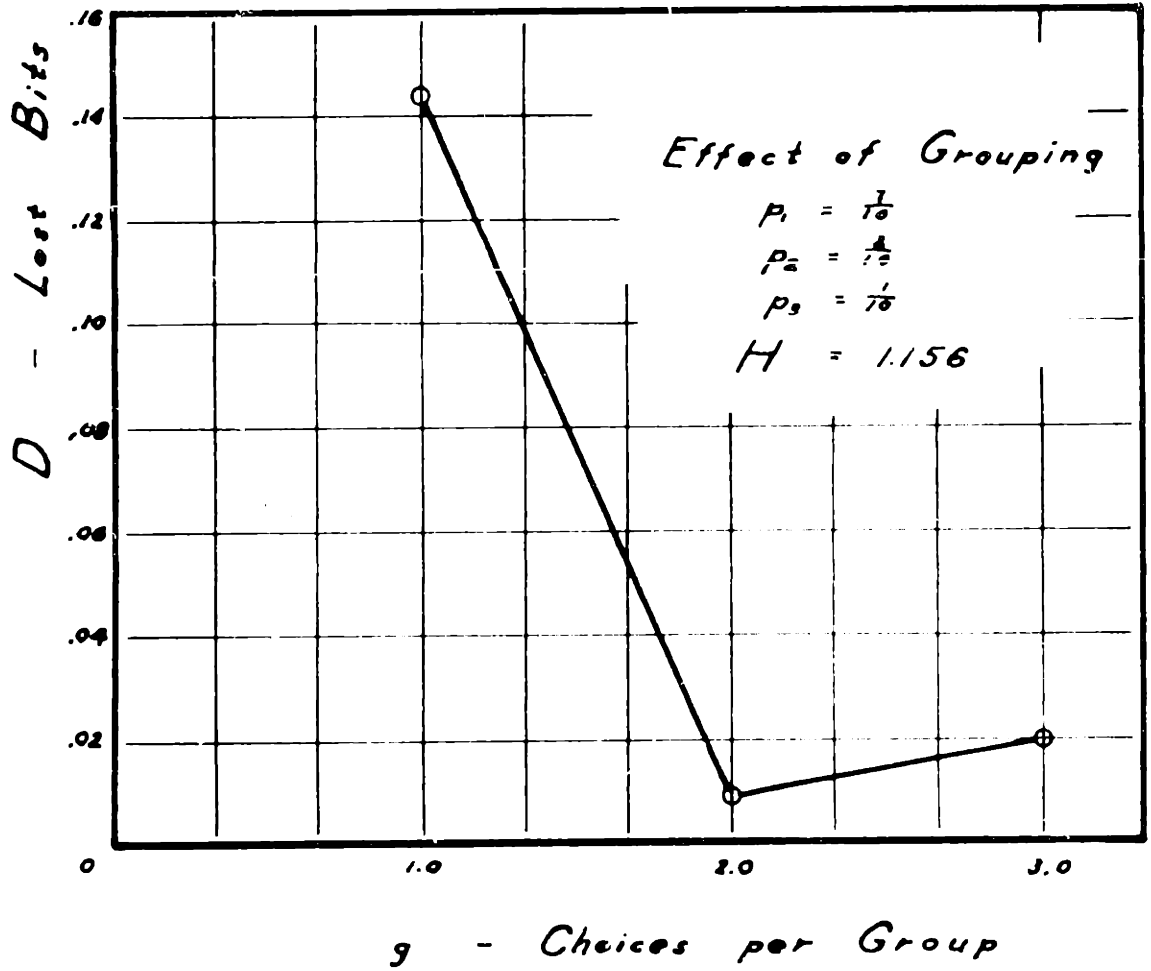


Figure 7

CHAPTER III

A DEVICE FOR CODING GROUPS OF PULSES

Introduction

It has been shown in the previous chapters that proper coding of groups of choices may reduce the average code length. This saving may be very important. For example, in storing information, it is usually desirable to reduce the message length in order to conserve the storage facility. Likewise, in transmitting signals, the shorter codes conserve time. For these reasons, it is interesting to consider the possibility of building a device for such coding.

This chapter describes a simple device involving a cathode-ray tube, mask, and photo-cell (and associated circuits), which, though severely limited in range, provides a means of grouping and coding pulses.

Description of Coder

Two somewhat different methods of operation, using the same equipment, were employed. In both methods, the cathode-ray tube, mask, and photo-cell were arranged in a manner quite similar to that employed by Dr. A. B. Macnee as a "Function Generator."[6] That is, the mask (a glass photographic slide) is placed next to the screen of a cathode-ray tube and the photo-cell placed in front of both so as to respond to light from the fluorescent screen that is not blanked out by the dark portions of the mask. The photo-cell output, a current proportional to the light coming through the mask, is converted to a voltage and amplified in a suitable amplifier. The output of the amplifier is applied to the vertical deflection plates of the cathode-ray tube in such polarity as to move the electron beam upwards, if a large amount of light is falling on the photo-cell, and downward in the other case. The result of this action is that the electron beam of the cathode-ray tube, if it were causing a spot of light not obstructed by the mask, is moved upwards until the spot of light reaches the edge of a dark area on the mask. At this position, the spot continues upward until an equilibrium position is reached where the light is partially obstructed, and just sufficient light reaches the photo-cell to maintain the position of the spot. With this arrangement, it is then possible to move the electron beam horizontally, and the spot of light will "follow" the edge of the black area on the mask.

In the function generator, the mask is of simple form. The edge of the black area is shaped to represent the arbitrary function. In the present device, the mask is more complicated and might appear as shown in Figure 8. More than one edge between an opaque and transparent portion of the mask are shown. Thus, with this mask, the equipment may be thought of as a multi-valued function-generator.

The multi-valued function-generator is used in the coder to perform the operations of quantizing, grouping, and coding. The signal to be coded consists of a series of amplitude-modulated pulses. These are applied to the vertical plates of the function-generator at a time when the spot is at positions (1) and (2), as shown in Figure 8. The first pulse at position (1) causes the electron-beam and spot to move up into one of the two transparent spaces shown above position (1). The feedback action of the light and photo-cell causes the spot to seek an equilibrium at the edge of a black area, as described above. The spot is moved horizontally so that when the next amplitude-modulated pulse arrives, it is in one of the two positions numbered (2) on the figure. The second pulse then moves the spot into one of the four transparent "slots." In this manner, the two signal pulses have been grouped to indicate one of the four slots. Although illustrated with a mask which allows only two levels to be distinguished and groups of only two pulses to be formed, it is seen that more complicated masks can be used. The fact that only a finite number of "slots" are possible at each pulse position "quantizes" the signal pulses or at least the effect of the signal pulses. The further use of the equipment to produce code pulses is simply to

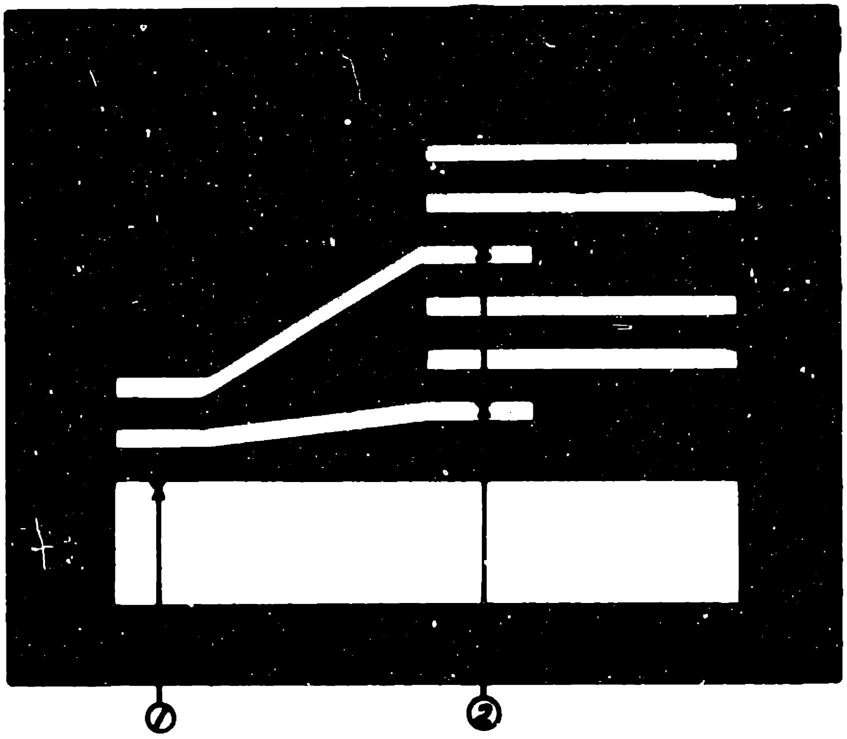


Figure 8

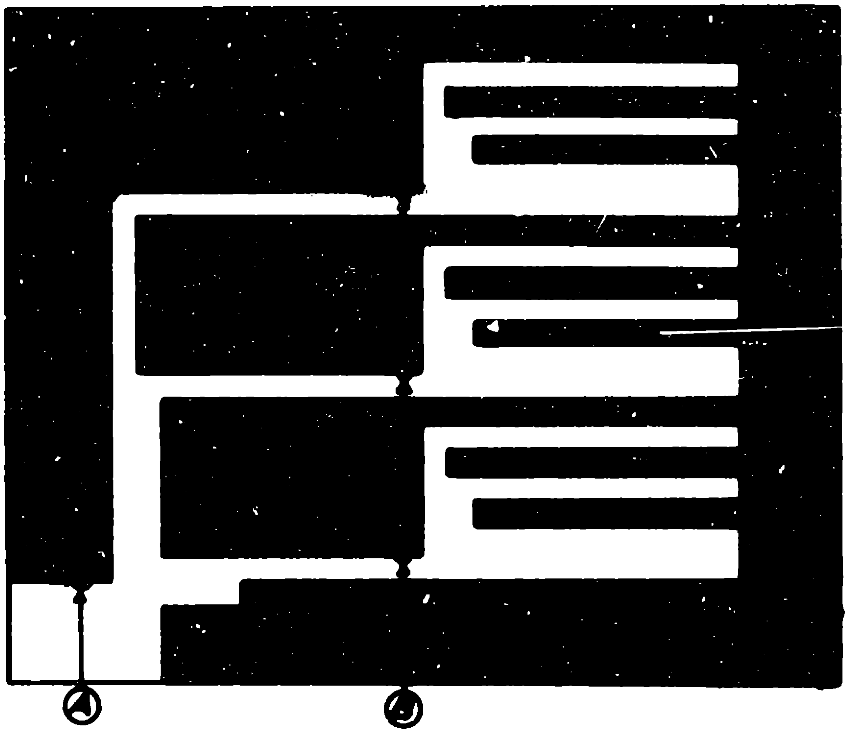


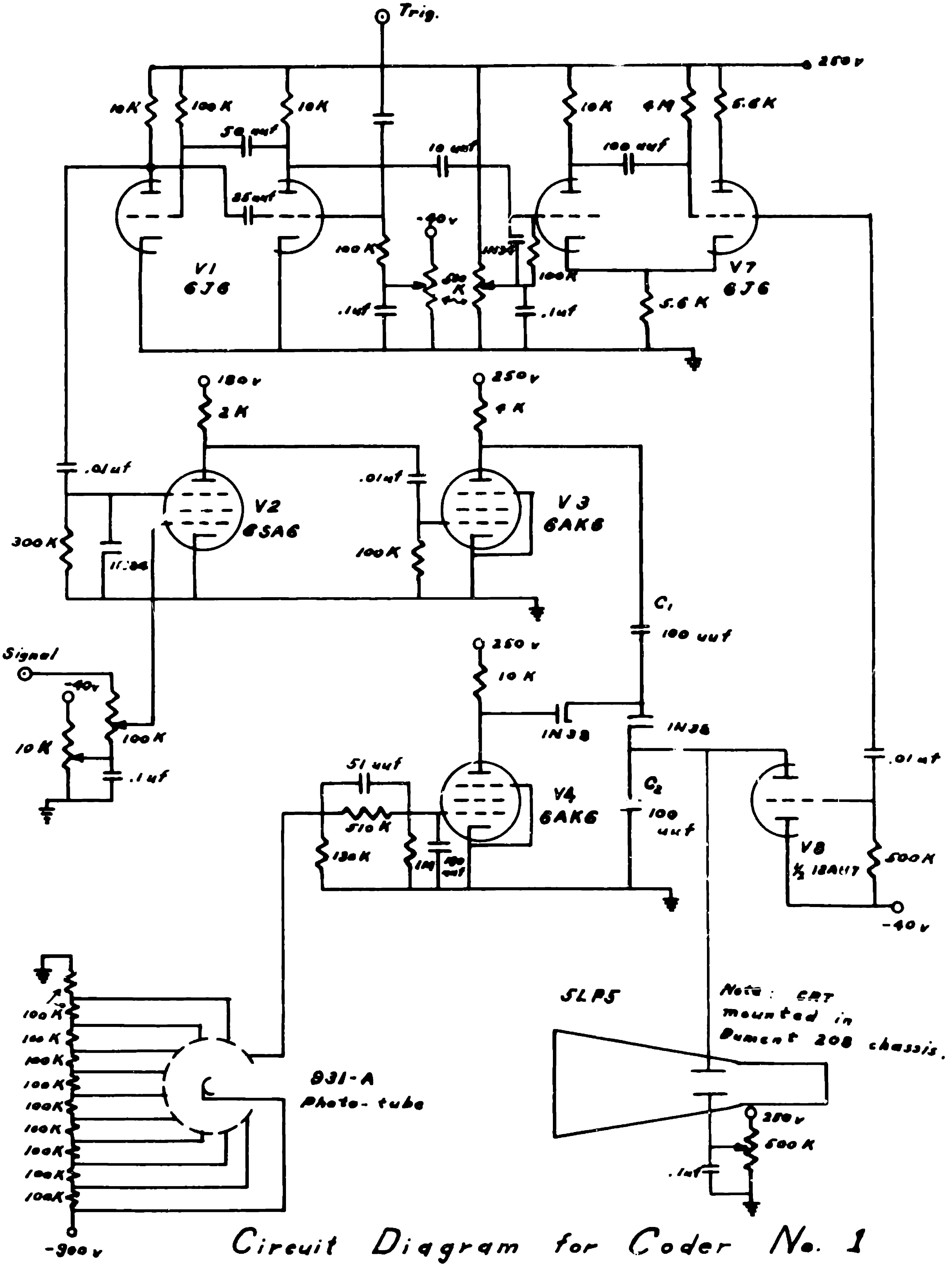
Figure 9

continue to move the spot horizontally in the long slots to the right of position (2). The edges of these black areas may be modified to produce code waveforms.

Brief Description of the Circuits of Coder #1

Figure 10 is a simplified diagram of the circuits used in obtaining the operation discussed above. V_1 is simply a multi-vibrator which causes short positive gate voltages to appear on the screen grid of the sampler tube V_2. If a continuous signal is fed to the control grid of V_2, its output will be a train of amplitude-modulated pulses. These are amplified and inverted in V_3. These three tubes thus produce a series of amplitude-modulated pulses.

The cathode-ray tube, photo-cell, and amplifier (V_4) comprise the function-generator, which operates to keep the spot positioned on the edge of a dark area of the mask. It will be noted, however, that the outputs of the two amplifiers (V_3 and V_4) are not fed directly to the vertical deflecting plate of the cathode-ray tube. Rather, the two voltages are fed through germanium crystal diodes to the capacitor C_2. The output of V_3', which is in the form of positive pulses, divides across C_1 and C_2 and leaves C_2 charged. This causes the spot to jump up on the mask, as described before. The output of V_4, which is a continuous voltage depending on the light incident on the photo-cell,



Circuit Diagram for Coder No. 1

Figure 10

may then raise the potential on C_2 in order to move the spot upward on the mask. However, the potential on C_2 may not be reduced, due to the diode. Therefore, the mask must always cause the spot to move upward. Figure 8 shows the mask is designed in such a way as to meet this requirement. Further, if the spot jumps to a position behind an opaque portion of the mask, it will remain there until the horizontal deflection has moved it far enough to "find" one of the transparent tracks.

The other circuits shown were designed to perform the following auxiliary functions. V_7 is a multivibrator which divides frequency, giving one output pulse for several input pulses from V_1. If the amplitude-modulated pulses are bein~ grouped by twos, this multivibrator would be set to give one output pulse for each two input pulses. The pulses from V_7 trigger both the sweep on the cathode-ray horizontal plates and the discharge tube V_8.

Several additional pieces of equipment are required for operation. A laboratory "trigger-box" starts multivibrator V_1 and determines the sampling rate. The cathode-ray tube, a 5LP5, is housed in a Dumont 208 oscilloscope cabinet, with the regular tube removed. This modified oscilloscope provides operating voltages for the cathode-ray tube and also the sweep circuits used.

Results Obtained with Coder #1

Unfortunately, the results obtained with Coder #1 were unsatisfactory to the author. Several rough masks were made, and only in the simplest cases did operation follow anywhere nearly like that expected. The pulse forming, modulating, and amplifying stages worked approximately as designed. The function-generator also performed reasonably well when tested on simple masks with only one black edge to follow. The following defects were observed:

(1) It was found that the function-generator was not able to hold the position of the spot in the narrow slots -- the overshoot of the feedback system causing the spot to jump too far.

(2) The single-ended deflection system produced serious defocusing of the electron beam. This changed the characteristics of the feedback system so that operation depended on the vertical position of the electron beam.

(3) The adding circuit, in which the feedback system could only move the spot in one direction, made the deflection system more sensitive to disturbances, especially those which might move the spot upward, since there would be no restoring voltage developed.

(4) The adding circuit resulted in interference between the signal-modulated pulses and the feedback system holding the spot position. That is, if the function-generator operates fast (as is desirable in following the changing edges of the black areas of the mask), it may cause improper voltage division of the signal pulses on C_1 and C_2.

Some time was spent investigating the troubles in this system and various minor modifications attempted. Although operation improved, it was never found to be stable, and the conclusion was reached that the system was inherently poor for the reasons enumerated above.

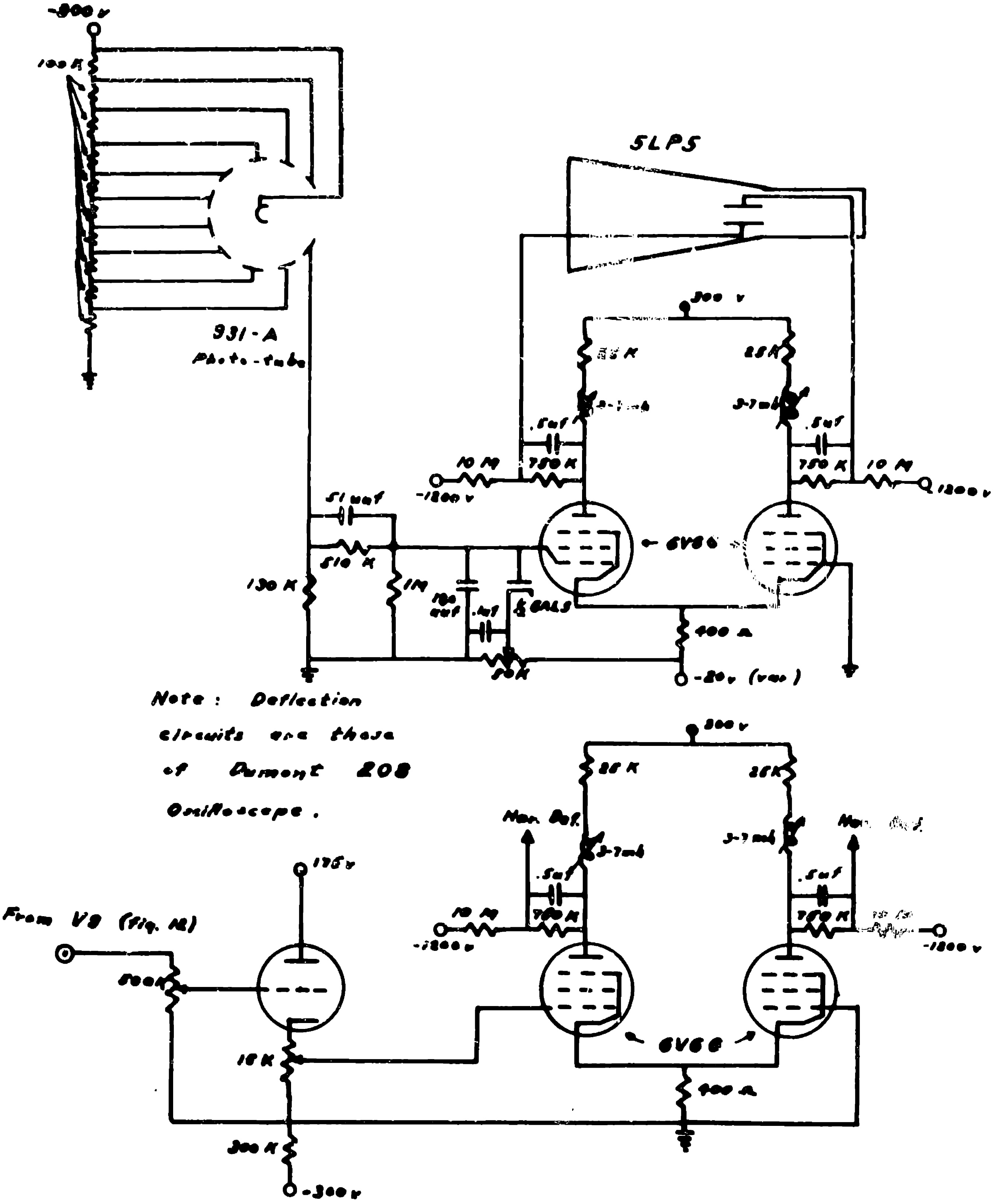
Description of Coder #2

In order to resolve some of the serious difficulties encountered with Coder #1, the system was redesigned and rebuilt, using the same equipment in a somewhat different manner. In this second arrangement, the function-generator feedback system operates independently of all other parts of the system, thereby enabling several circuit improvements. The function-generator in this arrangement performs the solitary function of positioning the spot of light on the edge of a transparent area on the mask.

The signal pulses are obtained as before but are now applied to the horizontal deflecting plates of the oscilloscope. A new type mask, illustrated in Figure 9, is used.

The cycle of operation starts with the spot at position (A) on this mask. The first amplitude-modulated pulse causes a horizontal deflection to the right proportional to the pulse height. The vertical deflection system then positions the spot against the mask by moving it upward. The spot is then moved to position (B), and the second signal pulse causes a second deflection to the right. When the spot has come to rest against a black edge again, it is swept across to follow the coded black edge.

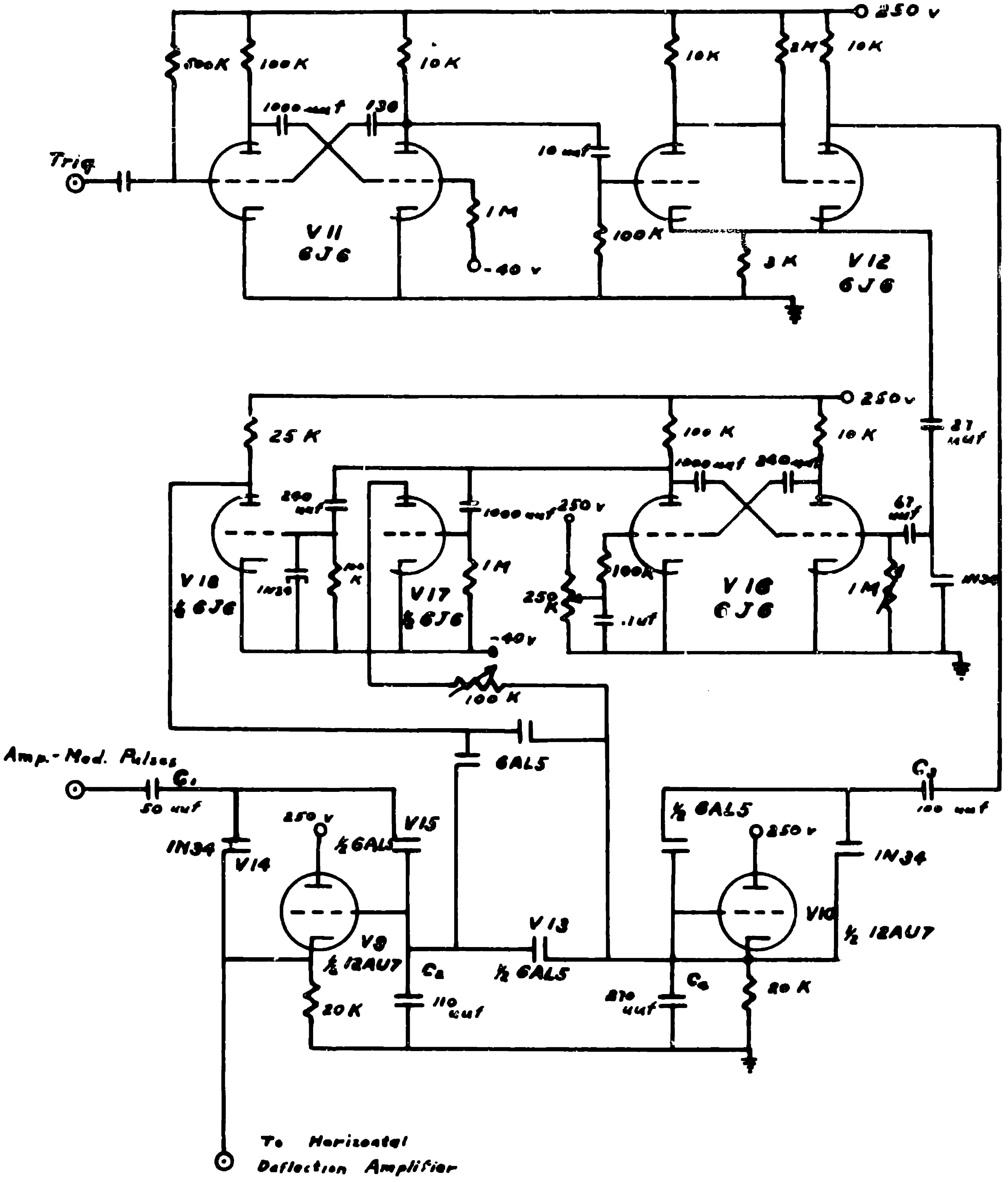
Figures 11 and 12 show the circuits of Coder #2. It will be noticed that the vertical deflection system is entirely independent of all other circuits. The cathode-ray tube and photo-cell are the same as in Coder #1, but the amplifier is quite different. The amplifier of the Dumont 208 oscilloscope, in which the cathode-ray tube is demounted, has been used to get push-pull operation at deflection plate voltages which will cause little defocusing. The gain is greater, and since the system will deflect both ways, disturbances have less effect than on Coder #1. The diode and potentiometer arrangement in the grid of the amplifier clip the input signal if it goes too far positive. This prevents deflection of the beam off the bottom edge of the scope and allows the phosphor of the cathode-ray tube to be operated in a biased condition. That is, a certain average amount of light is always present. The light may therefore increase or decrease towards its



Note: Deflection circuits are those of Dumont 208 Oscilloscope.

Circuit Diagram for Coder No. 2

Figure 11



Circuit Diagram for Coder No. 2

Figure 12

maximum or minimum value at about the same rate. The large gain in the amplifier means the light change for full deflection may be small, and therefore the system responds more rapidly. Operating in this manner, the time constant of the phosphor, which is normally about 25 us, may be reduced to about 2.5 us. The phosphor light build-up and decay are still the limiting factors in the response of vertical position systems.

The same circuits as were used in Coder #1 produce positive amplitude-modulated pulses, and these are fed to the circuit of V_9 on Figure 12. This circuit is a type of step-generator and operates as follows. The cycle starts with C_2 charged to a high value of voltage (due to other circuits). When a positive pulse is applied to C_1, its leading edge is differentiated in the short time constant of C_1 and the 1N34 diode (V_{14}) in series with the low output impedance of the cathode follower V_9, leaving C_1 charged. The trailing, negative-going edge of the pulse disconnects V_{14} but results in a division of charge across C_1 and C_2 through the 6AL5 diode (V_{15}). The voltage on C_2 thus "steps" down an amount proportional to the amplitude of the pulse fed to C_1. Since the input impedance of the cathode follower is high, the C_2 voltage tends to remain fixed. The next pulse results in another step down. Each time this occurs, however, the cathode follower sets a different level for the charging of C_1, and the net result is the subtraction of a voltage proportional to each amplitude-modulated pulse.

The output of this circuit is fed to the horizontal deflection circuits of the cathode-ray tube. Here again the amplifiers of the Dumont 208 chassis are utilized. This deflects the spot from the position (A) on the mask.

After a short time (about 25 us), the vertical deflection system has moved the spot against the black portion of the mask, and it is necessary to move the spot horizontally to position (B). This is accomplished by the circuit of V_{10}, which is similar to that of V_9. Positive pulses of equal magnitude are fed to this circuit from V_{11} and V_{12}. These two stages are multivibrators which simply serve to introduce the 25 us delay. When this pulse occurs, the voltage across C_4 steps down and, acting through the associated cathode follower V_{10} and the diode V_{13}, reduces the voltage on C_2 to the same level as C_4. The values of the capacitors C_3 and C_4 are chosen so that the voltage step on C_4 moves the spot to position (B) on the mask. This process may be repeated to form groups of more than two.

The remaining tubes perform auxiliary functions. V_{16} is a multivibrator which runs at a sub-multiple of the sampling frequency. It is triggered by a pulse from V_{12}, so that after the spot has been moved to position (B), it causes a positive gate on the grid of V_{17}. V_{17} then conducts, discharging$\overset{C_4}{/}$and causing the spot to sweep across the coded portion of the mask. At the end of this multivibrator gate, a negative trigger is applied to the charging tube V_{18}, which cuts it off, allowing both C_2 and C_4 to be recharged to a positive voltage.

Results Obtained with Coder #2

Coder #2 proved able to follow a mask such as that shown in Figure 9 with a sampling frequency of about 10,000 cycles per second. It is seen that, if 25 us is allowed for stabilizing the spot at position (B), 25 us for the code time (which seems a minimum if the spot is to follow the code edge at all), and 25 us more for setting the spot back at position (A), that there must be a minimum of 75 us between sampling pulses. This limits the sampling frequency to about 13 kilocycles.

The overshoot of the spot is still the limiting factor in how close the various slots on the mask may be placed. If the overshoot is reduced, the spot will not position itself as quickly, and the sampling rate would be less for reliable operation.

Some disturbance due to 60-cycle pick-up on the equipment was noticed. This disturbance results in effectively introducing a sixty-cycle modulation on the sample pulses.

Suggestions for Further Study

It has not proved practical, in the time devoted to this research, to exhaust completely the possibilities of such a device as described above. Nevertheless, it is believed that for slow speed sampling (less than about 8 kilocycles), and where only a few codes result from the grouping (less than about 50), that a device similar to that described might perform creditably. Some refinement of the present circuits to improve those deficiencies noted above should suffice for this.

The use of separate cathode-ray tubes for grouping and for coding would help to extend the range of sampling frequencies to 20 kilocycles or more.

In the case where a high sampling rate seems necessary and it is desired to group a considerable number of pulses, the problem is a severe one in switching techniques. Still, it seems important enough to warrant thorough investigation.

A closely related problem to coding is always that of de-coding. This problem may sometimes be solved by inversion of the coding equipment, but this need not always be so. An investigation of de-coding methods will be essential if coding of the type described in the first two chapters is to be exploited.

BIBLIOGRAPHY

1. Tuller; W. G., "Theoretical Limitations on the Rate of Transmission of Information," E. E. Doctor's Thesis, M.I.T., June, 1948.

2. Shannon, C. E., "A Mathematical Theory of Communication," BSTJ, Nos. 3 and 4, Vol. 27, July and Oct., 1948.

3. Wiener, N., Cybernetics, Chapter III, "Time Series, Information, and Communication," John Wiley & Sons, N.Y., 1948.

4. Fano, R. M., "On the Transmission of Information," Technical Rep. No. 65, Research Laboratory of Electronics, M.I.T., 1949.

5. Davenport, W. B., "Amplitude and Conditional Probability Distributions of a Quantized Time Function," Quarterly Report, Research Laboratory of Electronics, M.I.T., Oct., 1948.

6. Macnee, A. B., "An Electronic Differential Analyser," E. E. Doctor's Thesis, M.I.T., Aug., 1948.

7. Goodall, W. M., "Telephony by PCM," BSTJ, July, 1947.

8. Meacham, L. A., & Peterson, E., "An Experimental Multichannel PCM System of Toll Quality," BSTJ, Jan., 1948.