FACULTÉ DES SCIENCES ET DE GÉNIE
FACULTY OF SCIENCE AND ENGINEERING

SIMULATION OF A DIGITAL IMAGE PROCESSING SYSTEM

by

S.G.S. Shiva
J. -P. Dion

Electrical Engineering Department
University of Ottawa

UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

SIMULATION OF A DIGITAL IMAGE PROCESSING SYSTEM

by

S.G.S. Shiva
J. -P. Dion

Electrical Engineering Department

University of Ottawa

A report submitted to the Dept. of Communications under contract No. OSP3 - 0300.

Project Officer: Dr P. Allard, CRC

January, 1974

Abstract

The image transmission problem is introduced as well as the more important digital processing techniques for image bandwidth compression given. These different methods are then examined as to their relative merits and disadvantages. A study of pel amplitude frequency distributions and pel difference amplitude distributions is then made, using standard digitized images. From these results, companded DPCM followed by negacyclic coding is chosen as the source coding scheme to be investigated. A companded DPCM image processing and transmission system is then simulated. The resulting pictures are presented, along with a proposal for a system incorporating a negacyclic codec.

Acknowledgment

The authors express their sincere thanks to Dr P.E. Allard of the Communications Research Centre for his close cooperation and helpful comments. Thanks are also due to Messrs P. Lewis and Y. Blanchard, both of CRC, G. Garon and P. DesMarais, graduate students at the U. of O., and L. Michaud of Consolidated Computer Inc., of Ottawa, for their useful comments.

List of Abbreviations and acronyms.

A/D	:	analog-to-digital
BER	:	bit error rate
BW	:	bandwidth
BWC	:	bandwidth compression
codec	:	coder - decoder
DPCM	:	differential PCM
D/A	:	digital-to-analog
ΔM	:	delta modulation
FT	:	Fourier transformation
GT	:	Good transformation
HSR	:	high-speed reader
HT	:	Hadamard transformation
KLT	:	Karhunen-Loève transformation
mse	:	mean-square-error
NC, ND	:	negacyclic coder, decoder
pel	:	picture element
PCM	:	pulse code modulation
RCVR	:	receiver
ST	:	slant transformation
STC	:	spatial transform coding
T-domain	:	transform domain
XMTR	:	transmitter

Table of Contents

Table of Contents (cont'd....) Page

1. Introduction :

As the complexity of a society increases, the amount of information required for preservation of its existence increases. This is in fact true of our highly technological society, where the increase in information is truly explosive, whether it be in the fields of commerce, law, sicence or even entertainment. This poses related problems of first storing this huge mass of data and secondly, that of, whenever required, efficiently transmitting parts of this information to appropriate users. The word "efficiently" is very important since if no effort is made to reduce the amount of non-essential information or redundancy transmitted, communications channels will soon become overtaxed nearly as fast as they can be established. This applies as significantly to data storage channels: if the redundancy in a given amount of data is reduced, then that amount of data will have reduced storage requirements, or equivalently, more data can be stored in a given storage area. In this report, the problem under consideration is that of reducing information transmission requirements for certain classes of signals.

The process of reducing the redundancy in an arbitrary signal will result in its bandwidth being decreased, or equivalently to a reduction in its transmission rate or its storage requirements. The phrase bandwidth compression (BWC) will be assumed hereon to imply all of the above considerations.

Much of the information flow in to-day's communications is in the form of visual data, the most obvious example being commercial television systems; others are closed-circuit television, spacecraft picture transmission systems and facsimile. The use of these facilities would tend to increase if these communication systems were more efficient, and hence transmission quicker and cheaper. Also, the hierarchies of available systems would proliferate; clearly not every user would be interested in say a television system transmitting 30 frames/s, the standard commercial rate, if the visual data to be examined is some fixed image. To make this even clearer, consider the difference in requirements for transmitting a sound motion picture and a series of slides with a running commentary. Obviously, the second case would require a much lower frame rate, and so it would be a waste of money and channel bandwidth to use a standard TV channel.

The type of users who may be expected to exhibit interest in such a "slow" TV system might be professionals needing to consult graphical data during a telephone conversation, or commercial concerns requiring copies of charts, documents, etc. The availability of reasonably quick transmission channels for visual data would enable for example branch offices to reduce the overhead required for storing much of their visual information since it could be located at a central office and easily accessed by all branch offices.

Visual signals are among the most information - rich and highly redundant signals known [1] . This is conceptually clear if we examine an arbitrary image: most of the information resides in the contours of objects rather than in the actual grey level of the object. Hence the areas of a picture exhibiting the least information, i.e. the areas of only gradual change in grey level, which comprise the major portion of most pictures, is highly redundant. This will be more thoroughly explained in section 3.

Thus image signals are naturally of great interest from the data compression viewpoint. Source coding can usually be applied to great advantage to image sources in order to reduce their redundancy and so to reduce their transmission rate (or equivalently, reduce the image bandwidth, or compress the data). This results in a much more efficient use of available channel bandwidth, and a reduction in information transmission costs. For reasons of convenience and cost, the actual redundancy reduction can be done digitally, in a software environment.

In an earlier report [2] the more important digital image processing techniques for BWC were examined from a theoretical rather than an engineering viewpoint. In this report, engineering considerations will be stressed in comparing various processing methods. From these comparisons and measurements of some basic image statistics, a system is described which has been simulated on a PDP-8 minicomputer at the University of Ottawa. Simulation was used instead of a hardware implementation since changes in overall configuration and in algorithms can more conveniently and cheaply be effected using software rather than hardware.

The short-term goal of this project is to develop a practical, digital image processing system which can be used for BWC of slow-scan television transmissions (in an analog form) through a narrow-band voice channel. This report includes a preliminary proposal for such a system. What is to be developed is essentially a system which, using a slow-scan TV unit as an analog image source, will effect an A/D conversion on the picture signal, digitally process the resulting digital signal for redundancy reduction, then reconvert it to an analog signal for transmission through an analog channel in as short a time as possible. This must be accomplished while retaining the original image quality. The image reconstructed at the receiver by accomplishing the reverse of the above operations can then be compared subjectively to the same image transmitted without any processing for bandwidth reduction. This subjective evaluation can of course be supplemented by more analytical image quality criteria, though no criterion has as yet been found which truly characterizes image quality.

The long-term goals involve taking an analog picture signal, digitizing it and processing it digitally for bandwidth compression, then transmitting the resulting digital image signal through a digital channel. This type of system should become more important, especially since the availability of digital hardware and channels is continually increasing while their cost is decreasing [3]. In this project, all digital processing will be implemented by programming in software on a mini-computer.

2: Image Coding for BWC

2.1: Some Image Characteristics

A picture, as considered here, can be defined as a monochrome continuous non-negative function f(x, y) defined over a finite region of a 2-D space* and in which the value of the function at any point is given by the brightness level of the image, i.e. it is a real-valued function of two real variables. In general this brightness level varies from black to white, with a continuous distribution of grey levels in between. This definition applies as well to graphic, two level (i.e. black and white only) images, such as maps, line drawings and printed matter, as to the more common half-tone image, such as black and white photographs. For convenience, all discussions are confined to images of dimension N x N, where N is a power of 2.

Since all the processing considered here is to be implemented digitally, the continuous image must of necessity be sampled in 2-D space at the Nyquist rate, and quantized in amplitude. The samples themselves are known as pels (from picture element). Henceforth by mention of the word image, a digitized image is implied.

Also, by image is implied a representative image of the ensemble of images under consideration. It is possible to design an image such that its statistics are very different from the average. Examples of such images would be intensity wedges, some checkerboard test patterns etc.

---

* Pictures requiring motion rendition (eg., television) can be described as real-valued functions of 3 real variables: 2-D space and time. These are beyond the scope of this paper.

Hence, since an image must be of finite size, f(x, y) will be non-zero over only a finite range of values of x and y. Also, since there exists a practical limit, say M, to the brightness of any point in an image, f(x, y) is bounded by M.

To the observer, an image consists of many large, low-detail or textured areas, and relatively few sharp, high-detail contours, which describe the edges of the low-contrast areas. Loss of texture information results in a coarse image containing abrupt jumps in grey level instead of a quasi-continuous gradation in grey level. This leads to the appearance of a sharp discontinuity where none exists: this is referred to as false contouring or quantization noise. This is much more objectionable than unstructured noise of the same value, since the eye is much more sensitive to structured than to unstructured noise. On the other hand lack of edge rendition results in a defocussed picture, which is also quite objectionable.

Hence, to get a good quality digitized (or PCM) image, the sampling must be done at least at the Nyquist rate, while the quantization must be fine enough to remove false contours and correctly reproduce edges.

2.2: The Bandwidth Problem:

The problems posed in the transmission of PCM images can now be examined.

Given an analog image signal of say, 4 MHz nominal bandwidth, this can be sampled at 8 MHz, and if 128 grey levels are allowed, then 7 bits/sample (or pel) are needed. Hence, to transmit the PCM image in the same amount of time as the analog image, the bit rate must be $8 \times 10^6 \times 7$ bits per second, i.e. 56 Mbps, which requires a minimum transmission channel bandwidth of 28 MHz. Hence by digitizing the image, we have increased its bandwidth requirements by a factor which is equal to the number of bits used to represent each sample. Then, why digitize? The reasons are many-fold, but one of the more important ones is the capability of implementing practically any type of processing, whether linear or nonlinear, on a digital computer while for example, only linear operations could be performed on an analog image by an optical system.

Digitized image signals thus take up a broad bandwidth, several times the bandwidth required to transmit the same analog image, while the greatest advantages to be gained, as far as transmission is concerned, are ease of extraction of the digital signal from noise, and the capability to regenerate the picture signal over and over again on long distance communication links, while minimizing the noise. Unfortunately, the price to be paid for this, a 7 to 9 times increase in bandwidth, is much too high. Hence, we want to transmit digitized images, but using a much narrower bandwidth; what is then required is a digitized image transmission system which is much more efficient than "straight" PCM, so that narrower bandwidths can be used, or equivalently, so that the total number of bits required to transmit the image will be minimized.

To see how this can be accomplished, a closer look at image characteristics must be taken. Information theory states that most communication signals convey information at a rate well below the capacity of their transmission channels. The excess capacity is required to accomodate the redundancy, or repeated information, which the signals contain in addition to the actual information. Removal of some of the redundancy would reduce the channel capacity required for transmission, which is equivalent to bandwidth reduction.

Existence of redundancy is particularly evident in pictures. Many areas of little or no detail exist in an image, where the pels have the same or nearly the same grey level, and so the pel-to-pel correlation is very high. Hence a statistical correlation exists between the various points in the 2-D plane of the image, which gives rise to a 2-D planar redundancy. The elimination of this redundancy would reduce the amount of information that need be transmitted, with a corresponding decrease in bit rate, but not in non-repeated information rate.

From the preceding discussion, it is conceptually simple to visualize a system in which all of the image redundancy is removed. At the XMTR it would consist of an image scanner, decorrelator and rate-equalizing buffer, this last to match the variable bit rate of the decorrelator output to the transmission channel rate. At the RCVR, a buffer would adjust the channel rate to the variable rate required by the correlator. The output of the correlator would then be a reconstruction of the original image signal from the scanner in the XMTR.

Qualitatively, if an image is again considered as consisting of many large textured areas and relatively few sharp edges or contours, it can be seen that there exists a high correlation among the pels in a given textured area, a high correlation among the pels that "trace" a contour, and low correlation between the area pels and the edge pels. Hence, on the average, a high correlation among pels can be expected to exist in images.

In effect, all BWC methods, no matter what they are called, effect BWC by reducing image redundancy, the method used being or not being information-preserving. Four basic methods exist to accomplish BWC: parameter extraction, adaptive sampling, redundancy reduction and encoding. In general, an image processing technique for reducing image transmission bandwidth will make use of more than one method in an effort to optimize the digital image transmission system.

What follows is a brief look at some of the more important approaches to BWC, with a view to stressing their relative advantages and disadvantages. It should be noted that the classification system used has been chosen for convenience only and does not necessarily correspond to any other system of nomenclature. Also, most of the methods described here have already been examined in much more detail in [4] . They will be very briefly described here, the emphasis being on their relative advantages and disadvantages. Another very important point to be covered is the increase in susceptibility of the processed picture signal to channel noise, since the large amount of redundancy originally present served to immunize the signal to channel noise to a certain extent. Certain techniques such as PCM and especially DPCM will be covered much more extensively than in [4] , as they apply more to the system to be discussed in section 4.

2.3 : Methods of Achieving BWC:

2.3.1. Spatial Transform Coding:

2.3.1.1. Description:

Spatial transform coding of images is an attempt to make use of the redundancy reduction offered by both statistically-and viewer-oriented (psychovisual) coding. The adjective spatial is applied here to transform coding to underline the fact that, when transforming an image, we are transforming in space, the 2-D plane of the image.

Transform coding consists essentially of two steps, the first of which linearly transforms the original set of correlated pels into another set of coefficients, which we would ideally want to be independent of each other, but which can be, at best, completely uncorrelated, The first step is the one which is dependent on the image statistics. The second step consists in quantizing each, or a subset of, the coefficients. Here, psychovisual coding comes in : the number of bits required to code the coefficients and the portion of the coefficients examined will be dictated by the subjective effect on the human viewer, if that is the ultimate use which will be made of the transmitted image. The ultimate use to which the encoded image is to be put will largely determine the coding scheme for this step. The scheme that produces the "best" image as far as the viewer is concerned will not necessarily be the one that produces the best image in the mse sense or for further processing, say for pattern recognition. In other words, optimizing a reconstructed image for subjective viewing may degrade it from the point of view of information content, making it unsuitable for anything except viewing .

Spatial transform coding (STC) can be considered essentially as a process of dimensionality reduction. The transformation merely changes the position of the frame of reference for the image with respect to the coordinates, so that the most important information occurs along one portion of these axes, making the points in the coordinate system more independent, i.e. reducing their redundancy, so that some coordinates will become of negligible importance, and so can be deleted. More practically, the bandwidth compression occurs as follows: in the original image all pels have more or less the same importance, with the redundancy appearing in the form of interrelationships between individual pels. After transformation however, all points no longer have the same importance, with the points carrying most of the "true" information tending to appear in a different part of the transform domain from the points due mostly to redundancy. If the transformation is chosen properly, the information-bearing transform values are many orders of magnitude larger than those caused by redundancy, and so these latter terms can be reduced by either zonal or threshold coding. All transformations covered are information- preserving, since their Jacobian is unity. They are also linear and reversible, otherwise no image reconstruction would be possible. Another advantage of the transform method of image coding is the inherent channel error immunity, for low error rates, which results from the averaging operation of the transform, since each intensity sample of a reconstituted image is a weighted function of all transform samples. This is all the more remarkable since most other methods actually decrease noise tolerance of the transmitted signal by reducing redundancy, that which gives PCM signals part of their noise immunity.

In practice, STC is achieved by use of a digital computer, which accepts a digitized image for processing and outputs the image transform. The rule by which this is accomplished will of course depend on the particular transformation used. The next step involves filtering in the transform domain, either threshold or zone-wise. This is also accomplished by a digital computer, and the fewer the number of T-domain coefficients transmitted, the greater the BWC, and usually, the lower the quality of the reconstructed image. At the receiver, the inverse transform is used to reconstruct the image. A general STC image transmission system is illustrated in block form in fig. 2-1.

Each of the two steps described above can be implemented in different ways: the first step can use either 1-D or 2-D transformation, and there exists a very large number of possible transformations, as well as many different methods of T-domain filtering for the second step.

2.3.1.2. Step 1 : Spatial Transformation

The main advantages of 1-D spatial transformation as opposed to 2-D transformation are its compatibility with existing line-by-line image scanning equipment, lessening of the high-speed memory requirements, and the possibility of on-line processing of the image, which cannot be done as yet with 2-D transforms. There also exists special-purpose hardware for implementing certain 1-D "fast" transformation algorithms.

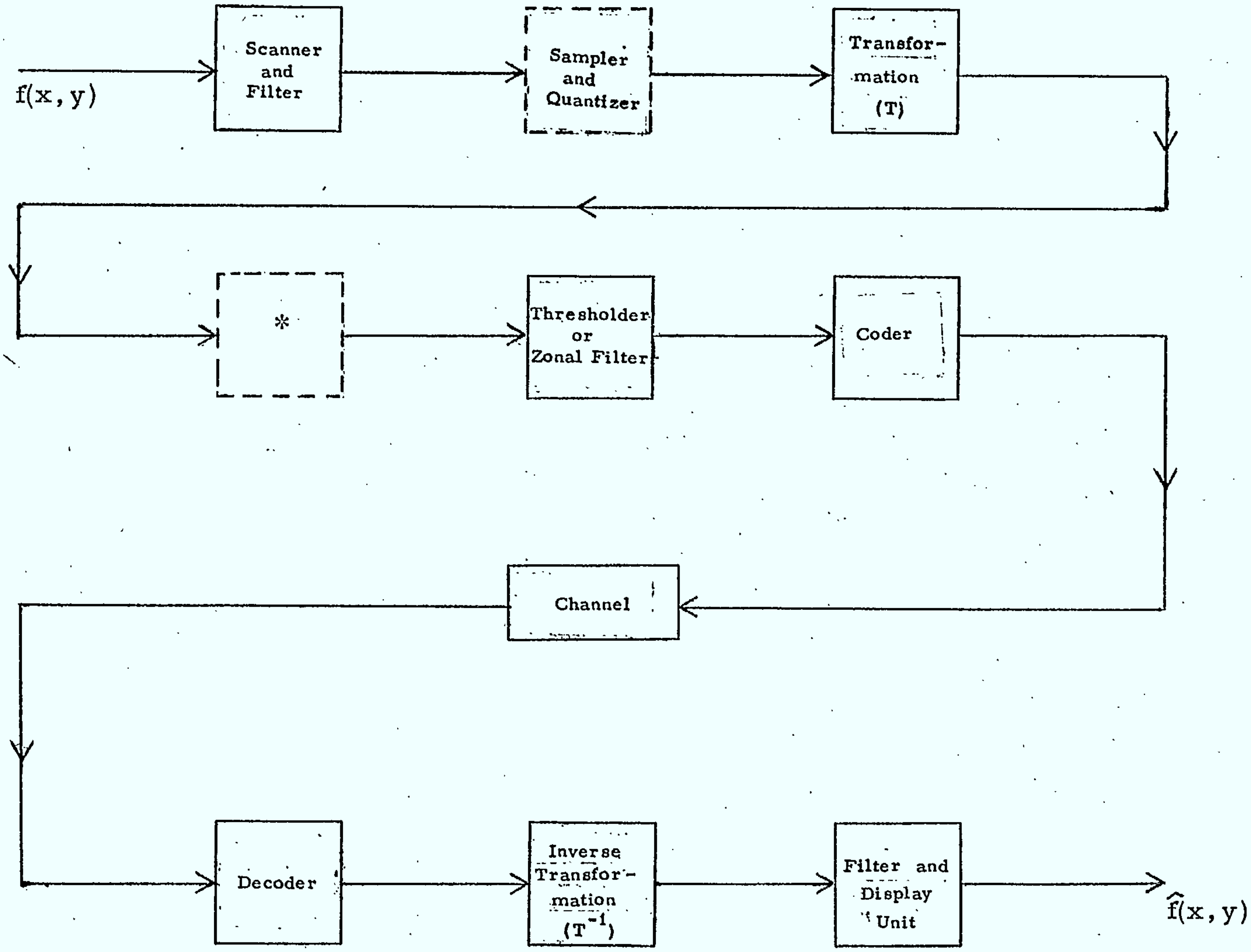


Fig. 2-I : Block diagram of a general coding image transmission system. The dotted block labelled * is an alternate location for the sampler and quantizer. In other figures, the filtering, scanning, sampling and quantizing functions are considered to be included in the scanner block.

Its main disadvantage is that, due to the fact it works on a pel-to-pel basis on a line, it cannot take into account line-to-line redundancy, and so less BWC can be expected than for the 2-D case. This is because results indicate that line-to-line redundancy (on a pel basis) is as important as pel-to-pel redundancy: clearly, why should image redundancy be a uniquely horizontal phenomenon ?

Some important 1-D transformations used in image processing can now be examined more closely as to their relative merits.

2. 3. 1.3 : <u>1-D Spatial Transformations :</u>

(a) <u>1-D Karhunen-Loève Transformation</u> (KLT)

This transformation is used principally as a yardstick against which other transform methods for BWC may be compared. This is because the KLT can be derived mathematically as the optimum transformation for a mean-square-error (mse) criterion. As will be explained further in section 2. 3. 1.4 (a), if a threshold is set in the transform domain, then the KLT is the optimum transform for minimizing the number of transform samples lying above the threshold while satisfying a mse criterion. The KLT is also known as the principal component, eigenvector, or Hotelling* transformation [5] .

---

* In an earlier report, the Hotelling and Karhunen-Loève transforms were treated separately. This may unfortunately have given the impression that they differ in some way. They describe the same transformation, with the same characteristics, but the KLT is usually defined on continuous variables, while the Hotelling transform is defined on discrete variables.

As for its drawbacks, even though it minimizes the mse for a given threshold, the mse is not a valid error criterion for many types of images, especially if they are destined to be viewed and not further processed. (Actually no good analytical criterion for image fidelity as far as human viewing is concerned is known as yet). Secondly, this transform requires a great amount of computation relative to other methods, and no "fast" transformation algorithm exists for the KLT. So the search continues for fast and efficient transformations, even though they may be suboptimal in the mse sense.

(b) : <u>1-D Fourier Transformation</u> : (FT)

The main advantage of this transformation lies in the availability of special-purpose hardware which will very efficiently compute the 1-D Fast FT (FFT), using say, the Cooley-Tukey algorithm, which can be implemented in $2\,N \log_2 N$ complex operations, where $N = 2^n$ is the side dimension of the square image. This results in savings in hardware costs. The fact that images are usually scanned line-by-line makes this a very tempting proposition for on-line image processing, even though to the author's knowledge, no such implementation has yet been realized.

A modification of the Cooley-Tukey algorithm that implements a 1-D FFT using $N \log_2 N$ complex additions and subtractions and only $\frac{N}{2}\left[\log_2(N-2)\right] + 1$ complex multiplications, has been developed. Since for most computers, MULTIPLY time is longer than ADD time, the time saving is significant.

Its main disadvantages lies in the fact that since the FT consists of complex numbers, all multiplication and additions must consequently be complex, and so are much slower than the same number of operations performed on real numbers. Hence it is slower than, say, the Hadamard transformation, which operates only on real numbers.

(c) : <u>1-D Hadamard Transformation: (HT)</u>

The main advantage of this transformation lies in the fact that special-purpose hardware exists to efficiently compute the 1-D Fast HT (FHT), using $N \log_2 N$ real additions, which is faster by about one order of magnitude than the FFT. Only additions are required since the HT consists only of $\pm$ 1's.

2.3.1.4: <u>2-D Spatial Transformations:</u>

For 2-D transforms, if the forward transformation matrix is constrained to be orthogonal, then the transformation can be interpreted as a decomposition of the image data into a generalized 2-D spectrum. Then each spectral component in the transform domain corresponds to the amount of energy of that spectral orthogonal function contained in the original image. In other words, the transform serves as an orthogonal basis for a decomposition into some generalized spectral representation. Hence the "usual" concept of frequency must be generalized to include transformation of orthogonal functions other than sine and cosine waveforms. Also all of the 2-D orthogonal transforms which will be examined will have the added requirements that their kernel be separable and symmetric . In practice, this means that the 2-D transformation can be implemented in two steps: the first step is accomplished by doing first a vertical row-by-row transformation, then doing a line-by-line transformation of the transitional 2-D data (or vice-versa). This is an important consideration in practice.

The main advantage of 2-D transformations for image processing is that they take both vertical and horizontal correlation into account. Their main drawbacks are the low speed with which they are implementable, due to the number and complexity of the required operations and the large-scale high-speed memory requirements. Following are the more important types of 2-D spatial transforms.

(a) <u>2-D Karhunen-Loève Transformation:</u>

This transform, which is nearly identical to the 1-D KLT, suffers from the same problems : computational complexity and lack of a fast computational algorithm. Since it is difficult to implement, but optimum, it is used nearly exclusively as a standard of comparison with other coding methods, and in particular, spatial transform coding methods. This transformation produces uncorrelated coefficients, minimizes the mse and packs the maximum amount of variance into the first k coordinates, for any k. ($k > 0$, $k \in I$) .

(b) : 2-D Fourier Transformation :

The main advantage of this transformation is with respect to 2-D eigenvector transformations : the FT has a fast computational algorithm, while the others do not. In block coding of images, the FT produces a smaller mse than the Haar and Hadamard transform for block sizes larger than 32 x 32, and also performs better than the slant transform for blocks greater than 64 x 64 .

However, the FT still has some disadvantages; for example, compared to the 2-D HT, which can be implemented using the same number of operations, but which consists of addition of real numbers rather than multiplication of complex numbers, there will be about an order of magnitude more time required for the Fourier transformation than for the HT.

(c) : Good Transformations : (GT)

This class of transformations is a proper subset of Kronecker Matrix Transforms(KMT) (i.e., so-called "powers of two KMT's"). One of their characteristics is that they are all implementable in $2N \log_2 N$ operations , i.e. they all possess "fast" computational algorithms. Following is an examination of the most important Good transforms .

(d) : Hadamard Transformation (or Discrete Walsh Transform)

This is one type of GT. Its advantage lies in the fact that it can be implemented in $N \log_2 N + N$ real additions , since the HT consists exclusively of $\pm$ 1's . Since for most computers, ADD time is much less than MULTIPLY time, the time difference in implementation for a FFT and a FHT is explained.

Hence the HT is faster in implementation than is the FT, and is also faster than the slant transformation.

Its main disadvantage is that the degradation due to the loss of high-frequency (i. e. "edge") information caused by zonal sampling is greater than for the FT, since it uses a rectangular wave basis instead of a sinusoidal one.

(e) : Slant Transformation : (ST)

This is the only transformation used for image transformation which was specifically designed to attempt to tailor itself to image characteristics and to computational efficiency requirements. It is an attempt to compact the image energy into as few transform domain samples as possible. It possesses a fast computational algorithm, and in at least one implementation, the ST gave a mse only slightly worse than that of the KLT and for some range of parameters, also slightly worse than the FT, but better than all others. For certain simulations, it was found to give subjectively much less degradation than the HT, and do only slightly worse than the KLT. No information on the time required for implementation has been given.

(f) : Haar Transformation :

This is an orthogonal but nonorthonormal transformation consisting of $\pm$ 1's and 0's which is directly related to the Walsh (or Hadamard) transformation. Its main advantage lies in that its implementation requires 2(N-1) operations, which is much less than 2 $N \log_2 N$ operations. Unfortunately the advantage of this very efficient computational algorithm is offset by its relatively large coding error, and so the Haar transformation has not generally been used for image coding.

More recently, [6] however, it has been found that the Haar transformation could give results which are subjectively as good as those obtained from the Hadamard transform, while the implementation is much quicker. The results are slightly better if block coding is used.

2.3.1.5: Basis Picture Interpretation of Image Coding:

Another way of referring to image transformation is to consider the transform as a series expansion of the N x N image onto say N x N basis pictures. Thus a picture is described as a weighted sum of "basis pictures". In practice, as when approximating a waveform by a Fourier series expansion, the series expansion is truncated after say k terms. Then, since the basis pictures are orthonormal, the mse of approximation by the truncated series is equal to the sum of the variances of the coefficients discarded by the truncation.

It has been shown that the Hotelling (or KLT) transformation in this respect

(i) produces uncorrelated coefficients

(ii) minimizes the mse of approximation

(iii) packs the maximum amount of variance into the first k coordinates, for any k.

Of course, for any practical-sized image, the N x N basis subpicture would require too much memory, so that a more feasible approach is to express an M x M subpicture (M < N, M/N) as a linear combination of M x M basis subpictures. It was found that the set of basis pictures resulting in the minimum mse was the Hotelling transform, followed by the Fourier, then Hadamard transform .

2.3.1.6: Step 2: Transform Domain Sampling

The techniques used for BWC in the T-domain can be classified as either being based upon the unique structure of the energy distribution in the T-plane or those that attempt to apply conventional spatial domain BWC methods to the T-domain rather independently of the energy distribution in the T-samples. The latter type has a much poorer performance due to the large dynamic range of the transform samples and their low (in one case, zero) correlation as compared to the original image pels which generally have a high correlation.

Many transform BWC techniques can be analyzed from the viewpoint of 2-D sampling, in which the image transform is multiplied by a 2-D sampling function which takes on the values zero or one ("dropped" or "retained") according to some a priori or adaptive rule. Then the reconstructed image consists of the original image plus some additive interference that is dependent upon the form of the original image and the sampling function. Both deterministic and non-deterministic sampling have been used.

(a) : Zonal Sampling

Since most images exhibit a high degree of correlation between adjacent pels, in the T-domain the greatest part of the energy tends to be clustered at certain spatial frequencies. Hence in zonal sampling, only those areas (or zones) of the T-domain which are expected to contain a high proportion of the energy are quantized and transmitted. This does achieve a large BWC but unfortunately, at the cost of defocussing: loss of high-frequency but low-energy T-samples results in an incorrect rendition of sharp edges.

(b) : Threshold Sampling :

To reduce this problem, T-sampling retains only those T-domain samples which are above a given threshold level. Unfortunately, though this method gives good picture reproduction, since we cannot predict which samples will be above the threshold, the position as well as amplitude of retained T-domain samples must be coded and transmitted. Hence the bit rate will be increased, thus decreasing the BWC.

2.3.1.7: Quantization Schemes for BWC:

This step is necessary to code the retained T-domain samples (after step 2) for transmission. Originally uniform quantization PCM was used to code the retained T-samples after zonal sampling but more recently, much work has been done on "matching" the quantization levels and quantum steps to the T-domain coefficient statistics, so as to achieve an even lower bit rate.

In the technique known as block quantization, a subsection (i.e. a subpicture) of the image is spatially transformed, a statistical parameter of the subsection pels is measured and the T-domain samples are quantized accordingly : where there is more detail, the quantization will be finer. The idea here is to achieve a high efficiency with respect to bit rate while maintaining the system complexity to a minimum; the cost in complexity for a very slight increase in BWC performance can be extremely high.

2.3.2: PCM Coding of Images:

Even though, as mentioned previously, PCM requires too large a BW for image transmission, it is still worth mentionning since it describes the basic or historical digital image transmission techniques. Its use is usually confined to being a standard of comparison for other image transmission systems.

PCM consists only of scanning, sampling and quantizing the image. (see Fig. 2-II). If the quantized samples are then coded and transmitted through a noiseless channel, an exact replica of the quantized image will be received; clearly, if the quantization is not fine enough, there will be some degradation in the received image, but only as much as in the quantized original image.

Given a restricted channel capacity (restricted to less than the Nyquist rate) and sampling the image at the maximum channel rate, the image will be undersampled,so a pattern of

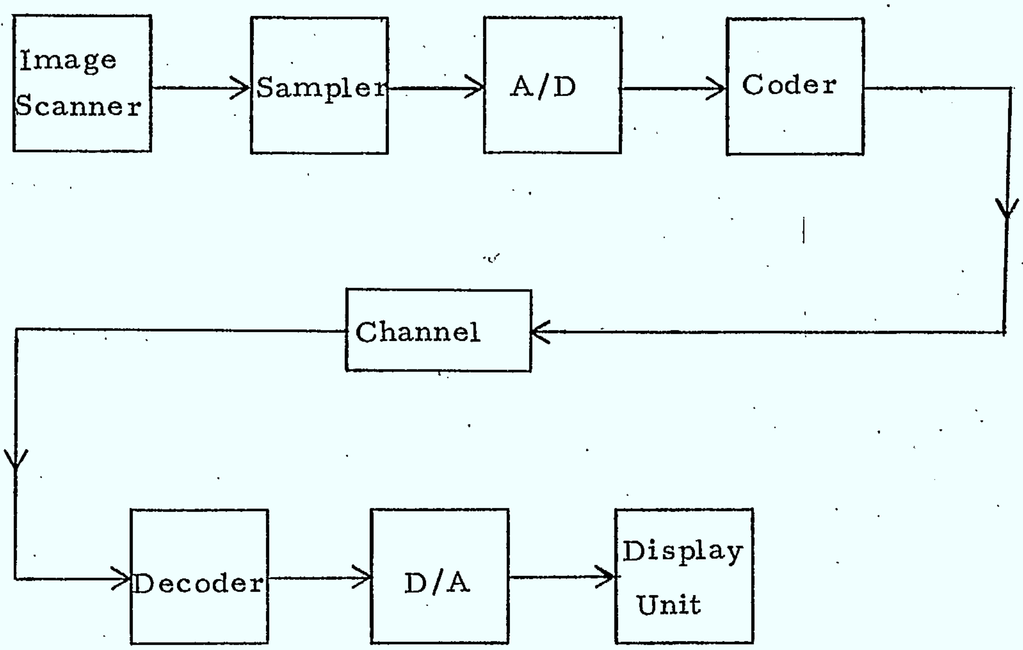


Fig. 2-II Basic PCM image transmission system. Unless otherwise specified, the D/A converter will be considered to be included in the block labelled "Display unit".

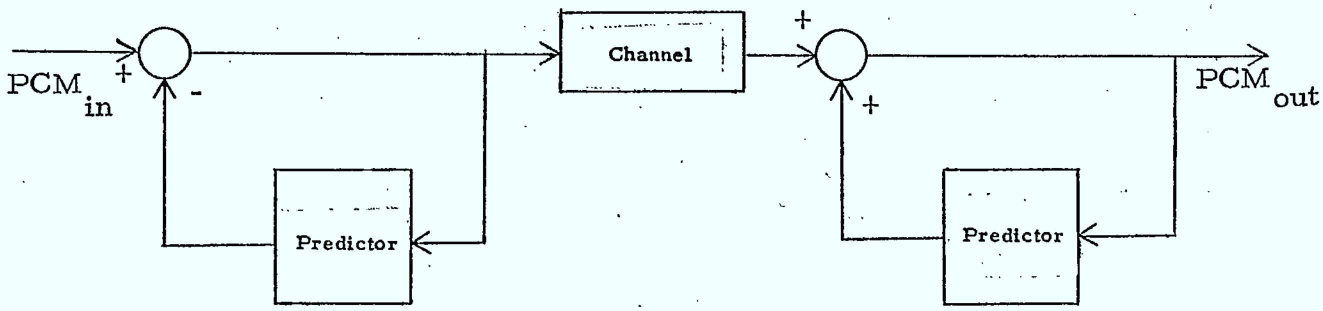


Fig. 2-III Block diagram of a basic DPCM transmission system.

dots and/or lineswill appear in the reconstructed image. This pattern is that of the sampling function used, and is often referred to as an artifact, since it is a structure which does not appear in the original image.

Another problem is that, if the grey scale is not quantized finely enough, false contours ("quantization noise") will occur in the reproduced image in the textured areas, i.e. where there is a gradual blending from one level to the next. These false contours are especially objectionable to a viewer since they are structured, and the eye is much more sensitive to structured than it is to unstructured noise of the same rms value. Use of this fact has been made in one scheme wherein pseudo-random noise is added to the analog picture before quantization, an identical noise being subtracted at the receiver; this serves to "break up" the false contours, permitting blending of grey levels in the reconstructed image. Unfortunately, while this method may be promising for images intended solely for viewing, it will of course degrade the image with respect to any further processing, such as for parameter extraction, etc.

Other attempts to arrive at breaking up false contours have been to use either random or ordered dither, which is a low amplitude high-frequency perturbation pattern: the penalty for this is a slight increase in the background noise level of the image. However, it does restore some of the information which a normal coarse quantizer would remove, such as spatial details and intermediate shades of grey. Again, this method should be used for viewer-oriented systems and not for information extraction.

2.3.3 : DPCM Coding of Images :

Since the human eye is much more sensitive to intensity differences than it is to absolute intensity values in images, and since there exists a strong correlation between nearby pels in most images, it is natural to assume that the transmission of pel differences as opposed to absolute pel values would afford a large BWC.

The basic difference transmission technique is differential PCM (DPCM), in which effectively the derivative of the input is transmitted rather than the instantaneous amplitude, as in PCM.

DPCM [7] systems are based primarily on an invention by Cutler. His original patent in 1952 was granted for a system which produced a quantized estimate of the next pel, given the previous pel or some combination of the previous pels, then transmitted the signed quantized difference between the estimate of the next pel value and the actual next pel value. His implementation to obtain the estimate, which he called a predictor, originally contained integrators; this explains why integrators are mentioned in the basic definition of DPCM.

Wiener [7] had derived the basic equation for linear prediction as early as the 1940's. In 1955, Elias [7] applied linear prediction to PCM coding, while in 1968 Graham [8] applied linear prediction to Cutler's basic DPCM system. Today DPCM is usually considered as a type of prediction quantizing system (see Fig. 2-III). Incidentally, PCM can be considered as a special (or degenerate) case of DPCM.

While it has been determined [7] that nonlinear prediction is superior to linear prediction in the signal-to-quantization error sense, it has not been determined how much better it would be, so that most DPCM image processing systems have confined themselves to the inclusion of a linear predictor in the quantization feedback loop. This makes analysis much easier, since nonlinearities introduce analytical problems.

DPCM as used in image processing is a system which gains a bit-rate advantage by making use of some statistical relationships of the image; it reduces the inherent redundancy in the image so that the transmitted signal consists mostly of non-redundant information. However, the very redundancy of PCM is what gives it such a large noise immunity; a one-bit error in a transmitted image sample will produce an error in one pel only in the reconstructed image. For the case of DPCM, however, there is less redundancy to provide noise immunity, and so this system is much more vulnerable to channel noise than PCM.

This is qualitatively obvious if we consider the worst case situation of one error occuring in the first transmitted difference only; then the first and all subsequent pels in the received image will be in error even if they are transmitted through a noiseless channel, since the 2nd pel is linearly dependent on the 1st pel, the 3rd on the 2nd, and so on.

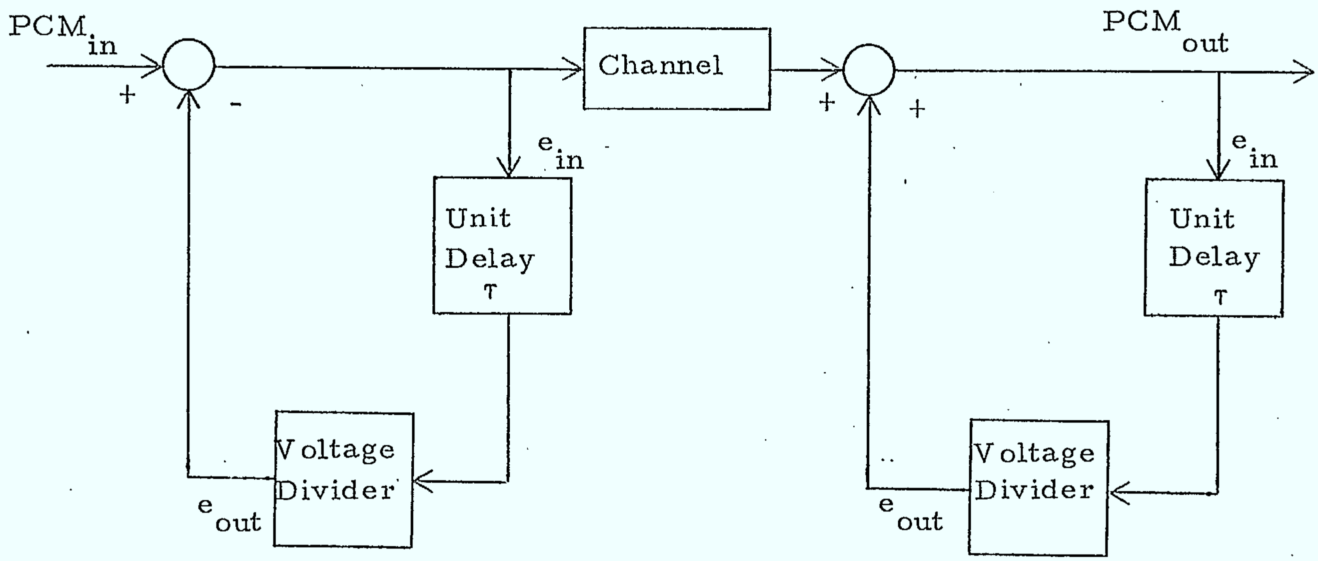
Clearly, these errors could affect at most one line of the reconstructed image if resets consisting of the actual (PCM) pel value were transmitted say,as the first pel on each line. Or, if the channel error rate is high, resets (or updates) could be sent more often, say three or four times per line. [9]

Also, it has been shown that [10] if DPCM is used to reduce the quantizing noise by say k db, then the error rate in the digital channel required for satisfactory transmission is reduced by a factor of $(1.26)^k$. However, this is of little consequence if the digital channel has a low error rate, and if the limiting degradation is that caused by quantization noise, then a decrease is quantization noise is desirable, even if this means increasing the noise introduced in the transmission medium.

In short, DPCM provides a reduction in bit rate as compared to PCM, or equivalently, an increase in image quality for a given bit rate, at the price of increased system complexity (and therefore cost) and a reduction in channel noise immunity.

Returning to the linear predictor as described above, in most cases the estimate is simply taken to be the difference between the current pel and the preceding pel multiplied by some factor ρ ($0 \leq \rho \leq 1$). This can be implemented very simply as shown in Fig 2-IV. ρ of course is the pel correlation coefficient.

However, there is no reason to believe that the above scheme is the best linear prediction scheme. Since there exists a high pel-to-pel correlation in images, the question is whether or not to confine the estimate to be a function of the previous pel only: over how many consecutive pels does a high correlation exist ?



$$e_{out}(t) = \rho\, e_{in}(t-\tau)$$

$$0 \leq \rho \leq 1$$

$$e_{out}(t) = \rho\, e_{in}(t-\tau)$$

$$0 \leq \rho \leq 1$$

Fig. 2-IV Block diagram of a DPCM transmission system using previous pel prediction. The predictor is a voltage divider whose setting is the correlation coefficient ρ.

Martin [11] made a study of the 2-D planar redundancy of images as a function of the number of "neighbouring" pels taken into account. He found that by far the greatest part of the redundancy is contributed by the first preceding point (on the same scan line). The only other pel offering any significant contribution to the redundancy for the pel under consideration is the one immediately "above" it, on the preceding scan line. Noting the relative importance of these two contributions, the case for the pel on the preceding scan line is weak, since the extra required memory would hardly be worth the expense, given the very slight increase in performance.

This is in contrast to the prediction used in DPCM speech quantization [12], where from 3 to 8 previous samples are used to predict the value of the next pel. It would obviously be a waste of hardware in this case, except in exceptional circumstances.

Unfortunately, Martin's results were for areas and did not specifically apply to pel-to-pel correlation along a scan line. However, Habibi has shown that, theoretically the prediction error from a n^{th}-order predictor decreases with n up to $n = 3$, with no appreciable change in prediction error for $n > 3$. His results applied to the $(n-1)$ preceding pels along a scan line. Hence predictors of order > 0 (i.e. utilizing more than the previous pel value in the prediction scheme) are useful for image processing. Fig. 2-V illustrates a generalized non-adaptive n^{th} order predictor.

Rice and Plaunt [13] state that correlation studies on Surveyor and Mariner pictures indicated that of all pels preceding a current pel on a scan line, only the one immediately preceding

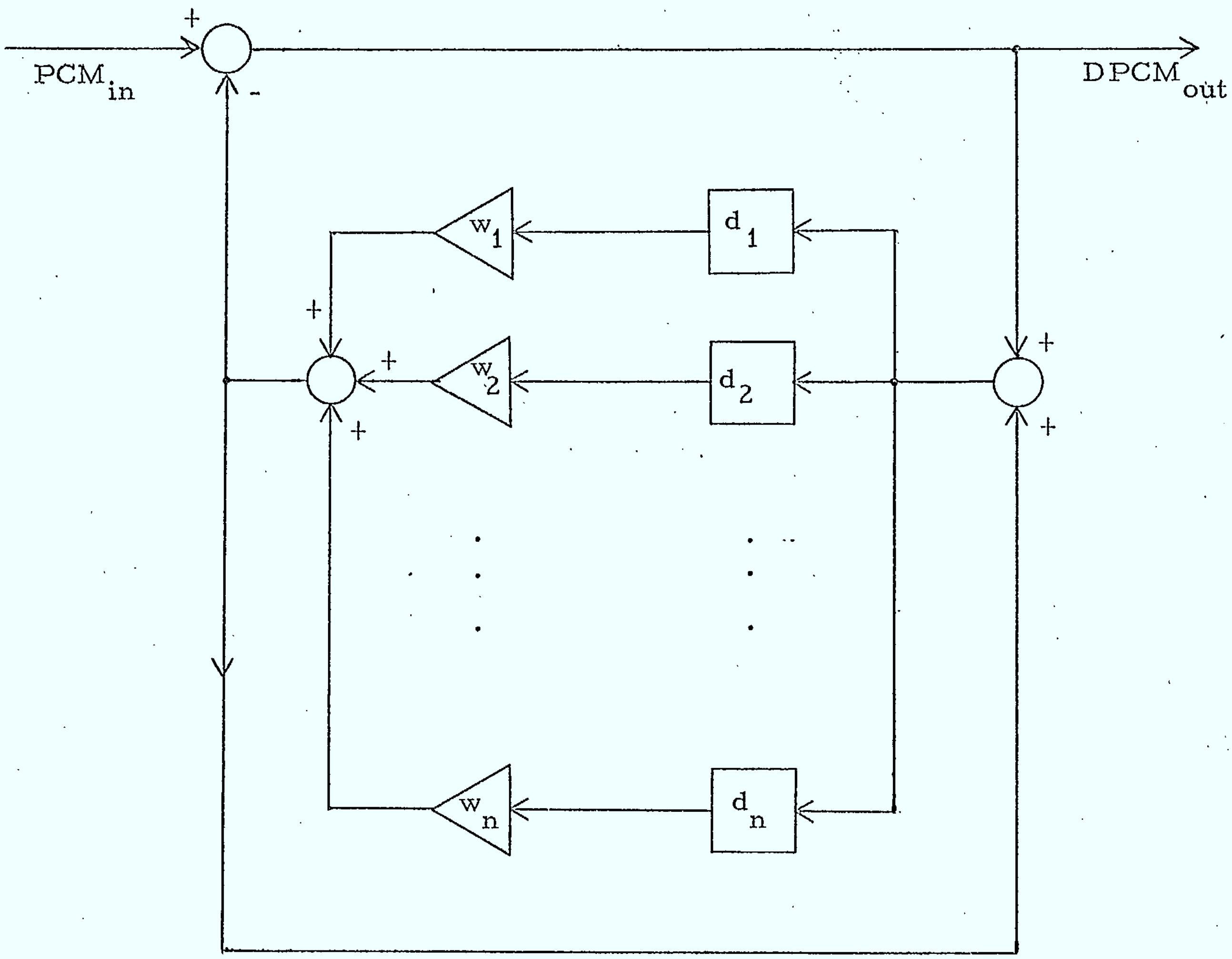


Fig. 2-V Block diagram of n^{th}-order linear predictive system (XMTR version). The w_i's are weighters and the d_i's are delays, with $d_{i+1} > d_i$ [7].

the current pel contributed significantly to the prediction process. Obviously, there is no general agreement about the order of predictor required.

Of course, if the prediction scheme is to be near-optimum, it must adapt itself continuously to local image conditions, which will substantially increase the predictor complexity. The adaptive part would be in the settings of the various weights or gains associated with each of the n previous pels to be considered.

So now, through the agency of some prediction and quantization scheme, a DPCM signal has been produced. Some of its characteristics can now be examined.

First, given a k-bit PCM input to a differential quantizer, the output will be a (k+1)-bit DPCM signal. At first glance, it seems we are producing bandwidth expansion instead of BWC. This is due to the fact that the absolute value of the pel difference can be the whole dynamic range of the gray scale, which requires k bits to express; however, the difference also has a sign, which requires an extra bit, hence the (k+1) bits required.

Fortunately, there is a way out of this [14] - this is due to a factor inherent in human vision. The sensitivity of the human visual system to small differences in luminance decreases at boundaries between light and dark areas. Thus, to match to this characteristic, small amplitude jumps are quantized finely while large amplitude ones are quantized coarsely. In other words, the DPCM quantizer must be a linear one followed by a digital compressor with a characteristic such as shown in Fig. 2-VI or the compression can be implemented in the quantizer itself. Whichever

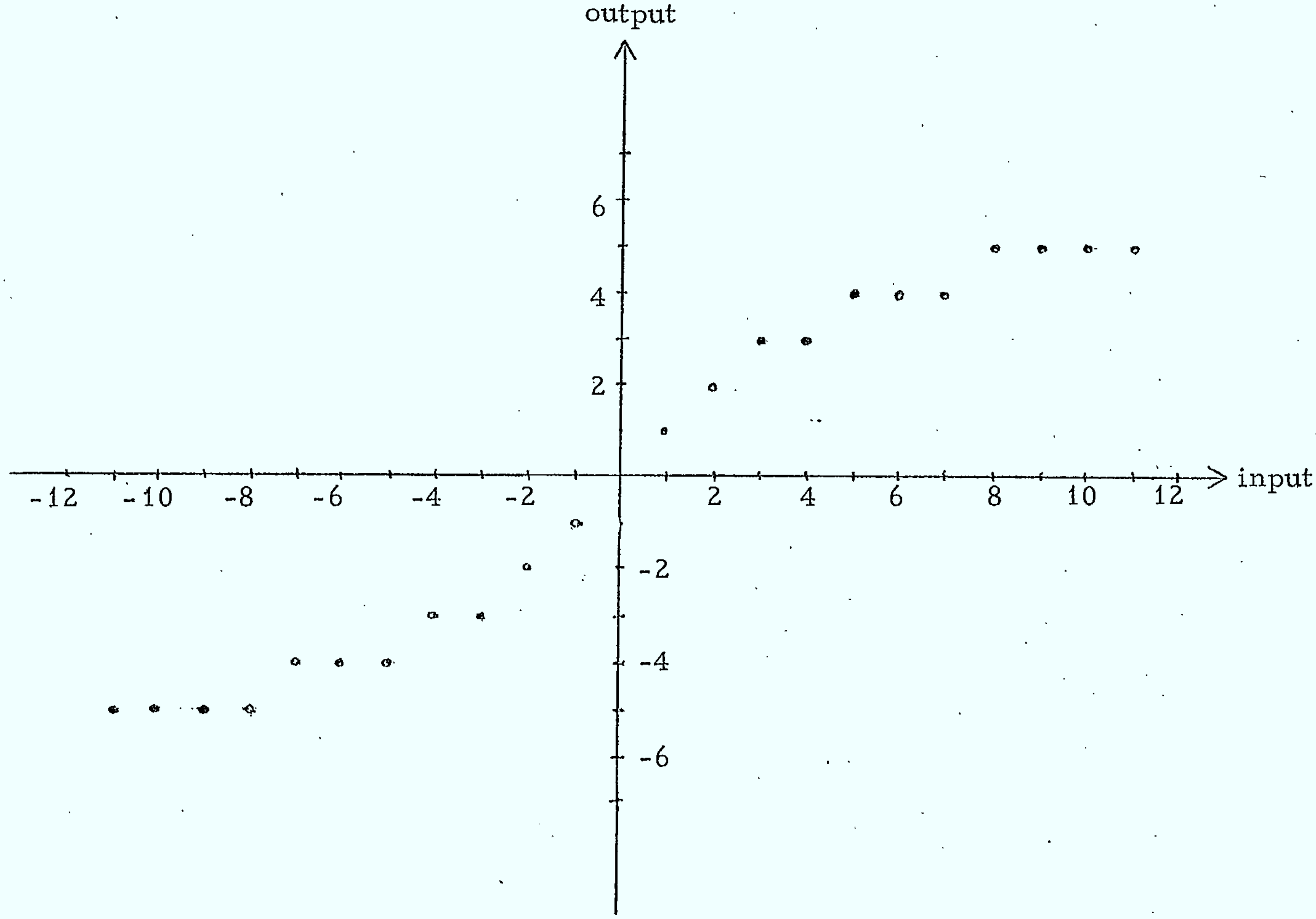


Fig. 2-VI A typical digital compressor characteristic.

method is used, the compressed DPCM signal is much more limited in dynamic range than with the original PCM or DPCM signal, hence the DPCM signal can be reduced to k' bits (k'< k). This will of course only produce an approximate reconstruction of the corresponding pel at the receiver, but it is possible to keep the reconstruction error to a low relative value, as will be seen below.

Another point of interest can be mentioned here; since in a DPCM signal, the differences tend to occur in bursts of the same sign, then the sign bit can be replaced by a sign difference bit. In other words the sign bit would normally be a 0 (say), and would be a 1 only if there were a change in sign from the previous sign. Clearly, this does not achieve any bit saving, but it does on the average reduce the weight of the DPCM words, if the average length of a one-sign run is greater than 2.

Another practical problem is the following: PCM usually contains an even number of levels, if all bit patterns are to be used for a given word length. Hence the DPCM words must express an odd number of difference levels (i.e. 0, $\pm$ 1, $\pm$ 2, ... etc). To remedy this situation, usually the 0-level, the odd level, is combined with either the +1 or -1 level, so as to produce an even number of difference levels. This will of course add some low-amplitude noise to the reconstructed image but it make more efficient use of the number of bits allowed for coding the DPCM levels. Clearly, this argument does not necessarily apply if, for k-bit DPCM words, not all possible 2^k words are used for coding the allowed DPCM levels. Then, of the left-over words, one might be used for coding the 0-level, and other words for special purposes, such as START SCAN and STOP SCAN commands.

The DPCM system as described up to now would have a configuration as illustrated in Fig. 2-VII.

Referring to Fig 2-VII the output $\hat{x}_i(t)$ for an input $x_i(t)$ can be considered. Say x_1 is entered, then x_1' is transmitted, and assuming a noiseless channel, x_1' is received, giving a reconstruction $\hat{x}_1$. Now generally there will be a difference between x_i and $\hat{x}_i$, and this difference will tend to increase on the average as i increases, assuming that the difference for the first image pel was 0. This is because of the nonlinearity inherent in the compression step after the quantization, so that most levels will not be exactly reconstructed. Of course, if the DPCM outputs a large jump, then the reconstruction error willbe large, but the relative error will be nearly independent of the magnitude of the jump. However, since large jumps appear relatively infrequently, these large reconstruction errors are rare. Also, errors will be introduced if the 0-difference level is included in either the +1 or -1 difference level; these last errors are of small amplitude (that of the smallest quantion level of the non-uniform quantizer) but they will appear very frequently. Hence, the effect of these two systematic errors is that, as i increases, even though the error in reconstructing a given jump may be unnoticeable, all these errors are being accumulated in the RCVR adder, such that on the average, the magnitude of this error can go on

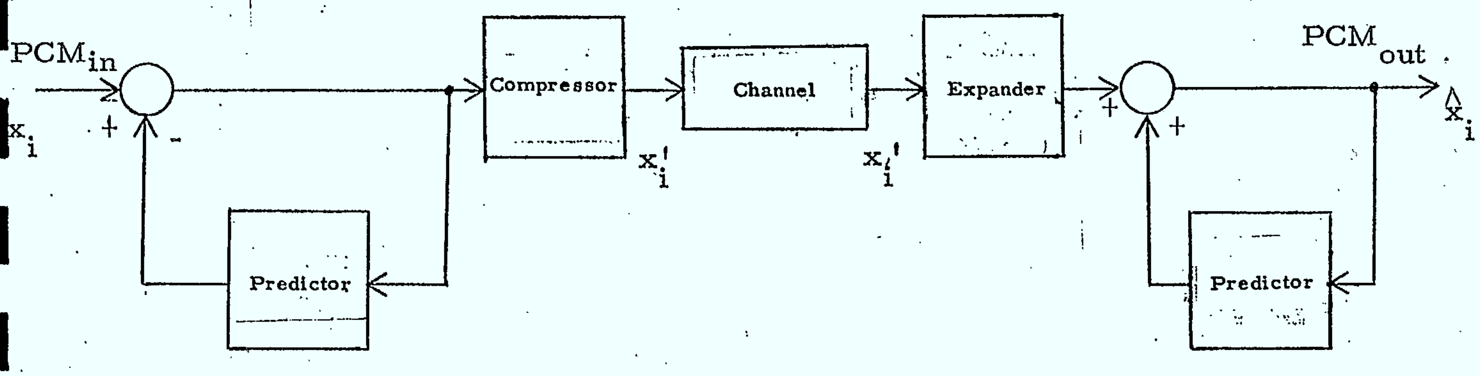


Fig. 2-VII Block diagram of a basic companded DPCM transmission system.

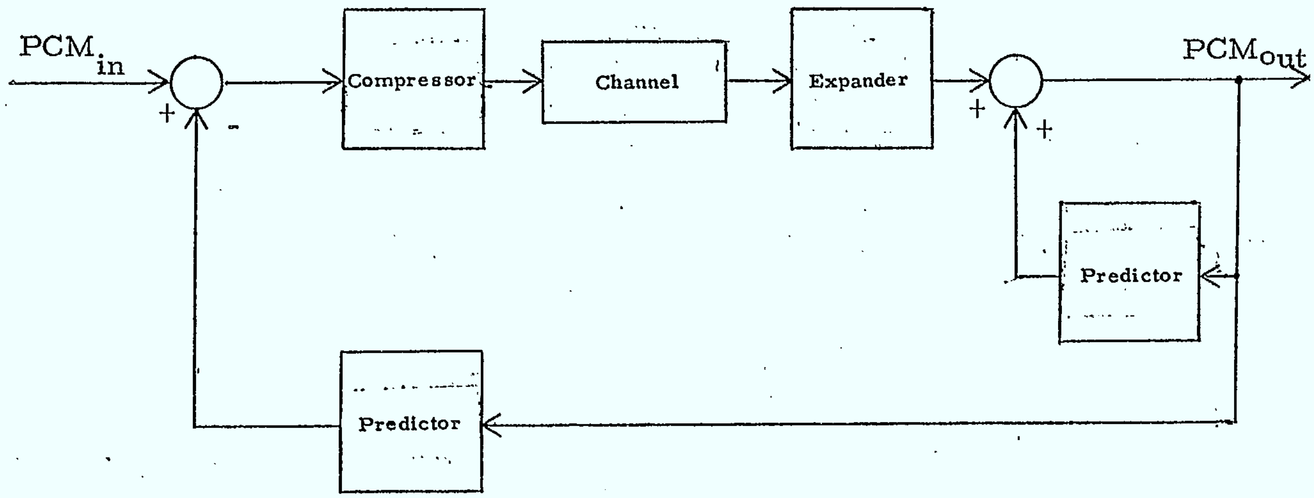


Fig. 2-VIII Self-correcting companded DPCM transmission system using feedback from the RCVR.

increasing. So the reconstructed image will at first closely approximate the original image, but as the number of transmitted DPCM words increases, the reconstruction error will on the average increase in magnitude, then decrease again, but often not without having caused long sequences of large amplitude errors, since the system is not essentially self-correcting. This effect can of course be limited to one line by resetting the first pel of each line to its PCM value, but may still be large for some portions of the line.

This problem has of course a simple solution conceptually, at the expense of more hardware. The DPCM coder must not use the actual previous pel(s) to form an estimate of the current pel, as there is no knowledge of the reconstructed pel errors and so the transmitted differences may tend to increase the reconstruction errors. However, if the difference between the previous reconstructed pel and the current pel are transmitted, then each new "jump" helps to compensate for the errors of previous jumps.

This is because, by finding the difference between the previous <u>reconstructed</u> pel and the current pel the next reconstructed pel will tend to be closer to the current pel, so that the system is essentially self-correcting, given a noiseless channel. Of course, due to the companding scheme used, this error will not always go to zero but it will severely limit the magnitude and recurrence of errors.

This system then utilizes feedback from the RCVR, and would have a configuration as illustrated in Fig 2-VIII. This system would seem to imply the use of a separate feedback channel to transmit the reconstructed values to the DPCM predictor. However, if the part of the RCVR which produces the reconstructed pel values is also incorporated in the XMTR, the reconstructed pel values will be available at the XMTR for its linear predictor. The overall system incorporating this feedback is illustrated in Fig. 2-IX.

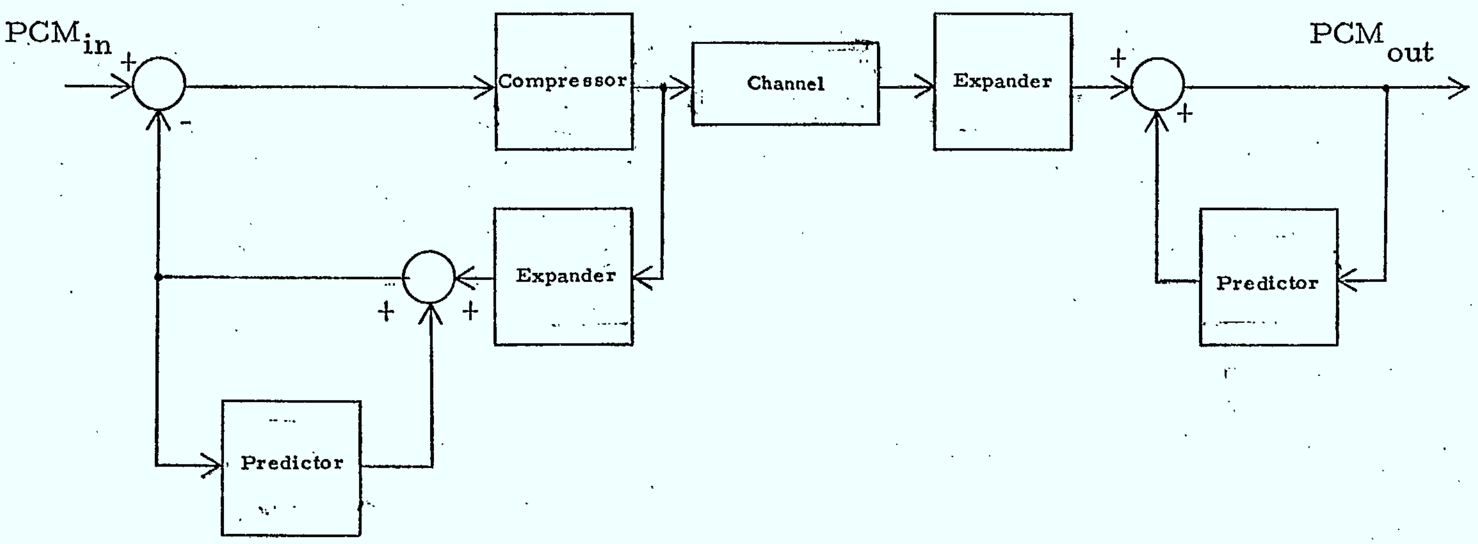


Fig. 2-IX: Practical self-correcting companded DPCM transmission system. Feedback from a replica of a portion of the RCVR located at the XMTR makes available the previous reconstructed pel at the XMTR .

2.3.4: Delta Modulation :

This coding system can be considered as a special case of DPCM, in which only 2 levels (± 1, usually) are allowed, and the linear prediction is the difference between the current pel and the previous pel. Clearly, this system is extremely simple to implement, and so, reliable and cheap.

Refinements to this system have been devised in which for instance, the step size depends on a weighting circuit, the step size chosen being a function of the number of repetitions of similar bits, such as would occur at an edge. The weighting circuit can be a function of some local average of the signal, or a function of the instantaneous slope of the signal.

Unfortunately, when using Δ M, for a given bit rate, the reconstructed image is nearly always inferior to straight PCM.

The principal advantage of Δ M systems are their simplicity. However, as just seen, the complexity of the system can be increased in such a way as to reduce the bit rate, and hence achieve better BW.C. However, some of these implementations are so complex that the simplicity advantage of Δ M over say DPCM is reduced or lost altogether.

The principal disadvantages of Δ M are the following : at sharp edges, the delta modulator cannot correctly track the edge, since it is restricted in step size; the solution to this is to increase either the sampling rate or to devise a system where the step size is adapted to local image conditions. The second problem is the presence of granular noise whenever the coder "sees" a string of equal-valued pels, since then the coder outputs alternate positive and negative steps. Also, for a given SNR, a higher sampling rate is required for Δ M than for DPCM.

2.3.5: Dual Mode Systems :

Nearly all of the systems previously considered suffer from either or both of the following; if the system is designed around proper edge rendition, requiring high spatial frequencies, then usually the low spatial frequencies will be badly reproduced at the RCVR, and vice-versa. An optimum system would consider edges and textured areas separately: it would sample the lows less finely, but quantize them more finely, and vice-versa for the edges. It may be possible to achieve this using adaptive sampling, but this has not been done as yet.

A dual-mode system is one which utilizes a different quantization procedure for different spatial frequency bands, usually separated into a "highs" and "lows" band.

The different general approaches have been two-fold: one is to directly transmit a low-definition, analog image for textured areas and to superimpose on it an image consisting of the edges or contours (contours are connected edges). The other is to code the "lows" and "highs" using different schemes, transmit them separately, and recombine them at the RCVR. To transmit edge information, both the position and the amplitude of the edges must be transmitted, usually using run-length coding. This results of course in a bit-rate increase.

Another approach has been to transmit the edges by locating and tracing all contours in an image. Edge points are detected, using some appropriate property of edges, such as the gradient, then finding other points of similar amplitude which are connected to it, thus tracing a contour. Of course, the positions of points in a contour must also be transmited. In other implementations, curves are fitted to some of the contour points to approximate it. From a psychovisual viewpoint, contours are very important since the human eye judges the intensity level of an area by means of the contour surrounding it it.

The main disadvantages of these systems is first, the fact that when only portions of an image are transmitted, the positions of these points must also be specified; secondly it has been found that all the information contained in an image cannot be reproduced using only a low definition "lows" image and a "contour" (or "highs") image. This is because edge information not located on a contour is lost if it is of low contrast when contours are drawn.

3: Image Statistics of Interest : [15], [16]

Examination of a representative image, say an aerial photograph of a city, poses the problem as to the specific location of the actual information and that of the redundancy. Eventually, an intuitive feeling is arrived at: the most important parts of the images seem to be the shapes, rather than the grey levels of objects in the image. The shapes are defined by connected edges, or contours, which represent large jumps in pel amplitude. This is well illustrated in the work of cartoonists, where the contours of the drawings carry most of the information. Since the information is mostly concentrated at the edges, the redundancy must also be distributed non-uniformly throughout the image: it is very small in high-information areas and vice-versa. Thus the redundancy at edges and contours will be small. Thus it is to be expected that little BWC can be achieved at edges, but most of it will occur in areas of low information, such as blank or textured areas.

Since the most relevant information is located in areas of abrupt changes in amplitude, it would seem normal to represent edges by a differential function, such as the Laplacian or gradient function, or more simply, the derivative of the function. Noise consideration of course preclude differentiating the analog image signal, but if the signal is digitized, the differences between successive quantized samples (i.e. "jumps in amplitude") can be considered, Clearly, large jumps or differences will correspond to sharply-defined edges, and in a typical image, these will be relatively few in number .

Hence the problem can be restated as follows: what is required is an efficient method of coding digitized images. To find an efficient method, we need some knowledge of the digitized image statistics. Clearly, if these statistics are about the same for all or most images, then some coding scheme adapted to the image statistics can be found.

One basic statistic would be the frequency distribution of amplitude levels (i.e. pel values) for a given image. This was done on a PDP-8 minicomputer for the three images available on perforated tape: MOON, GIRL and COUPLE. The resulting distributions are illustrated in Fig. 3-I. The actual numerical values for the distribution plots are given in Appendix A. Clearly from Fig. 3-I, these three images have very different amplitude frequency distributions. Hence if a Huffman code were designed to match to MOON, say, it would give a BWC for MOON, but it might actually give a bandwidth increase for GIRL and COUPLE. So this statistic is of no use for image coding.

If the frequency distribution of pel values is of no help, then some other statistic might be found to be nearly stationary, i.e. independent of the image, or nearly so. Since, as previously mentioned, most of the important information is in the pel differences, it is natural to examine the frequency distribution of differences between pel values. Here, for simplicity, only the differences between adjacent pels is considered. Also, at this time, it is convenient to introduce a parameter ρ, such that the i^{th} pel difference Δ_i is given by

$$\Delta_i = pel_i - \rho \times pel_{i-1} \quad , \quad 0 \leq \rho \leq 1 \qquad (3\text{-A})$$

in which ρ is the correlation coefficient, an estimate of the correlation between the i^{th} and $i-1^{th}$ pels.

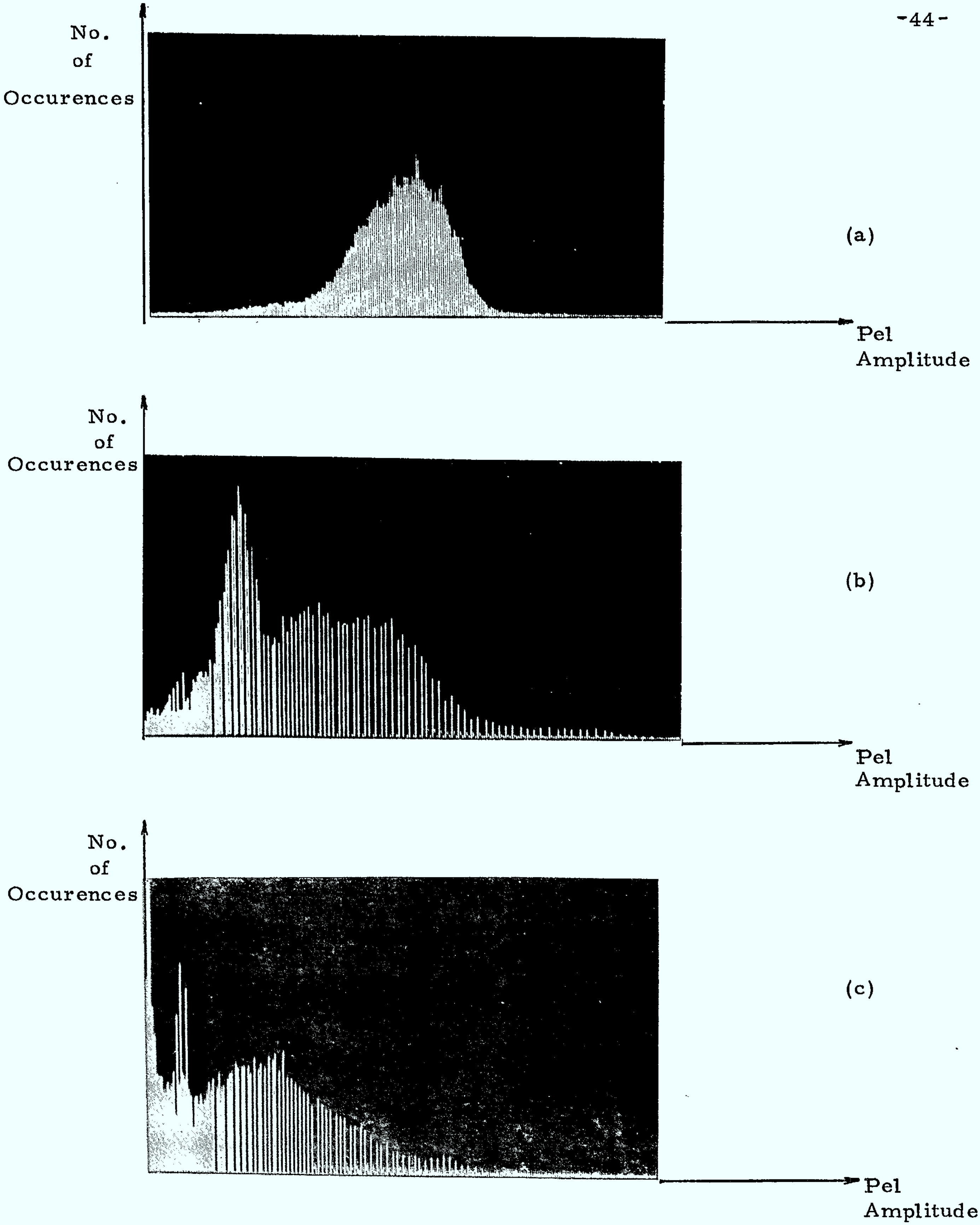


Fig. 3-I : Frequency distribution of pel amplitudes for MOON (a), GIRL (b) and COUPLE (c).

Using (3-A), the frequency distributions of the pel differences for MOON, GIRL and COUPLE were measured. The graphical results are presented in Fig 3-II, while the numerical values are given in Appendix B.

It can be readily ascertained from an examination of Fig. 3-II that, for a given picture, the statistics are heavily dependent on ρ. One limiting case is for $\rho = 0$, which gives the same distributions as in Fig. 3-I. However, the most interesting case occurs for $\rho \cong 1$; this gives a frequency distribution which appears very nearly gaussian, centered at the zero-difference point. This is consistent with previous discussions of image properties and the high correlation which has been found to exist between adjacent pels. It also agrees approximately with the results obtained by Estournet [15].

Hence the probability of a zero pel difference is high, while that of a large difference is small. But what is of utmost importance is that, for $\rho \simeq 1$, all three images give about the same pel difference distribution. So pel difference statistics are quite similar for all three images and this fact can be used in the design of an efficient image coding scheme. Obviously, this scheme will be some type of DPCM system.

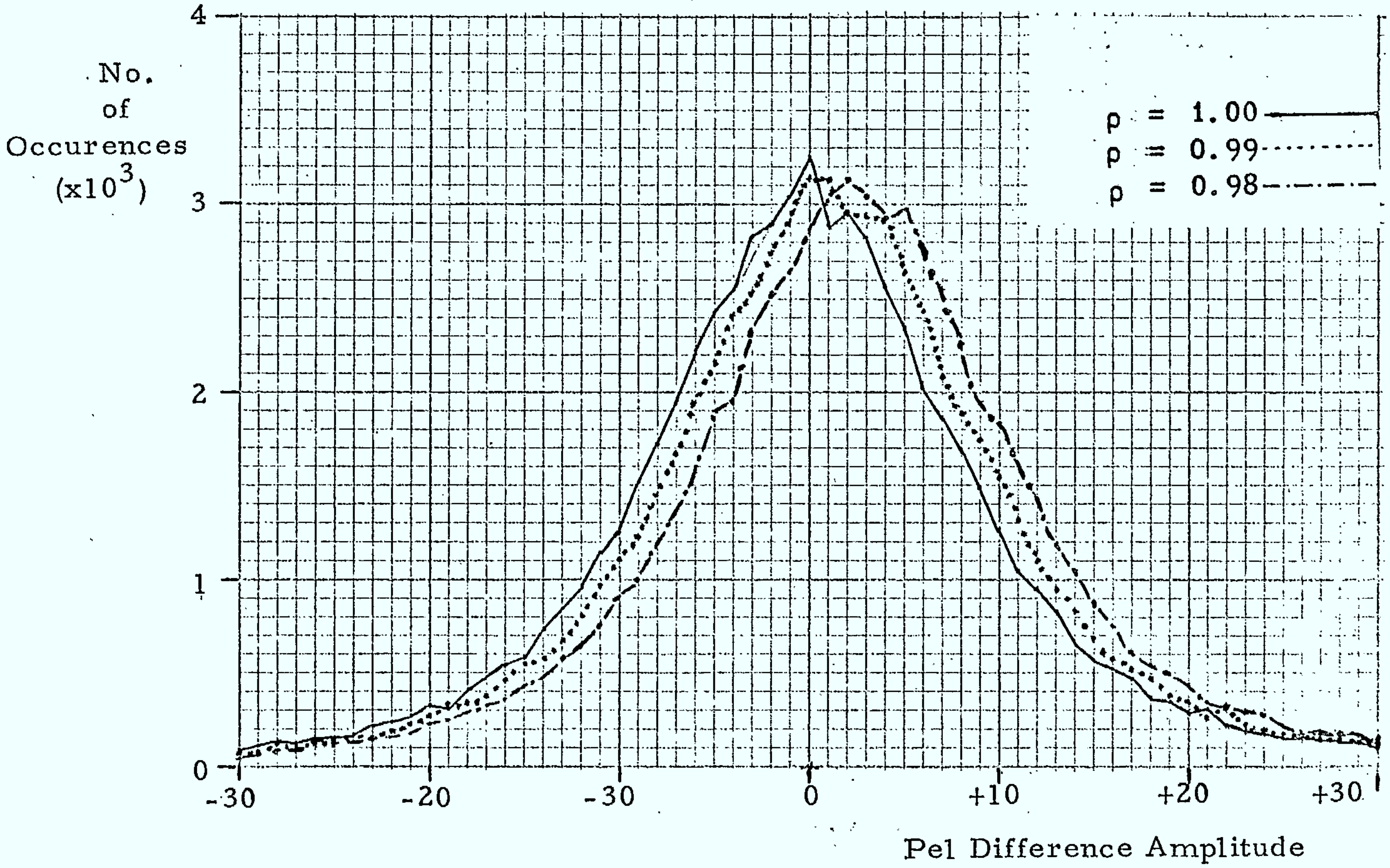


Fig. 3-II (a) : Frequency distribution of pel differences for MOON, with ρ = 1.00, 0.99 and 0.98.

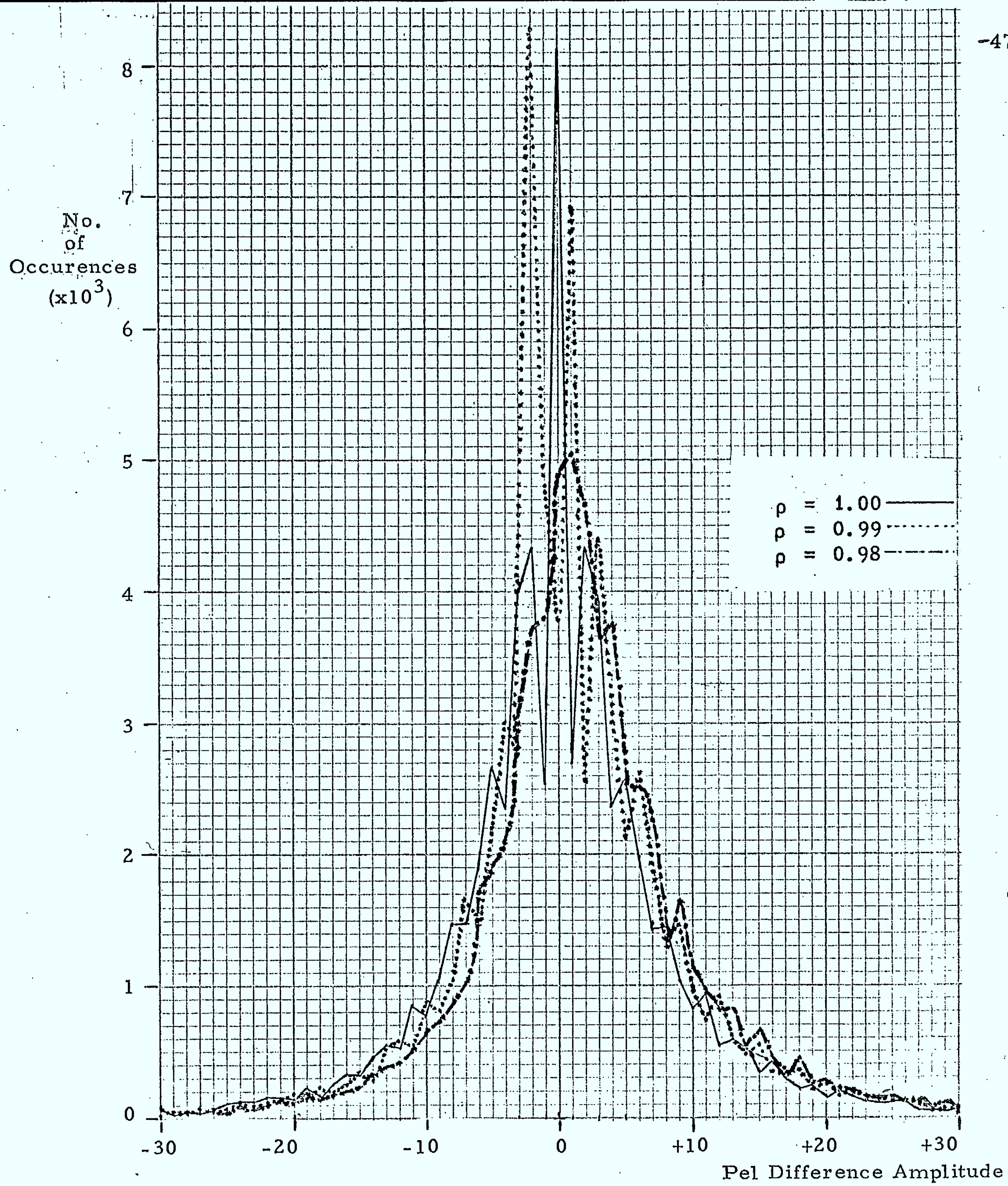


Fig. 3-II: (b) Frequency distribution of pel differences for GIRL, with ρ = 1.00, 0.99 and 0.98.

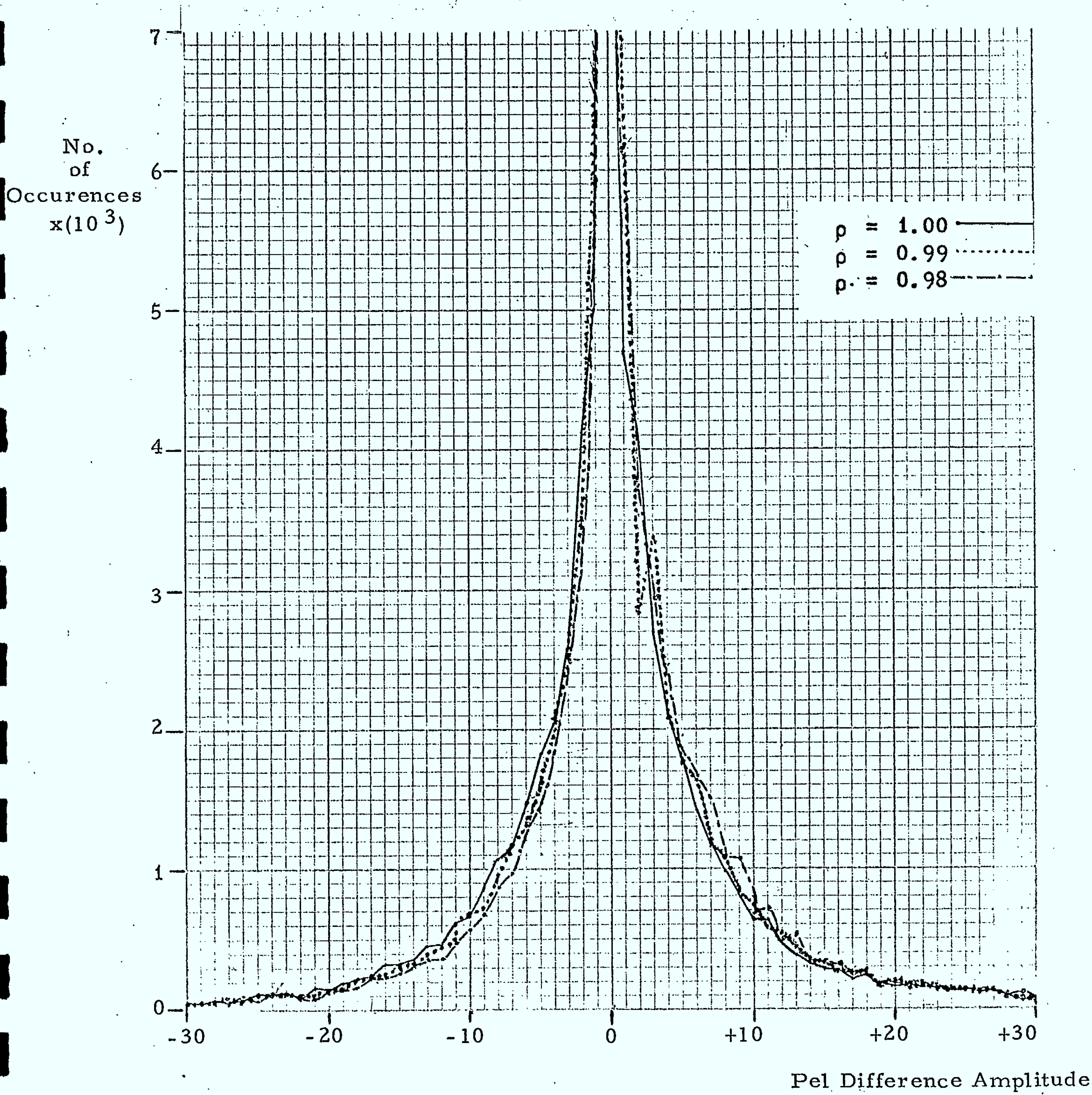


Fig. 3-II'(c) Frequency distribution of pel differences for COUPLE, with $\rho = 1.00$, 0.99 and 0.98.

4. Measurements and Simulations:

Work has begun at the University of Ottawa on simulation of a companded DPCM* system which takes into consideration the points discussed in section 2.3.3.
A DEC PDP-8 minicomputer was used for the actual implementation, with a high-speed paper tape reader (HSR) for reading image tapes, while a D/A converter was used to output images on a CRT display unit. All three sample images, MOON, GIRL and COUPLE were available with each pel stored as an 8-bit binary number on perforated paper tape. Thus the pels exhibited a range of levels from (00 000 000) to (11 111 111), or equivalently, from 0_8 to 377_8. This results in 400_8 (= 256_{10}) possible grey levels for the pels. The image dimensions were all 400_8 x 400_8, resulting in $65,536_{10}$ pels. Thus the original images are described by $(256_{10})^3$ = 16, 777, 216_{10} bits.

The procedure arrived at was for an image processing program to be loaded on the computer, then one of the image tapes read in on the HSR. The pel values, after processing, were then reconstructed and outputted on the CRT display. The resulting image was photographed using either a Polaroid Land camera or a 35 mm Praktika LTL camera fitted with a 1.8/50 lens. In all cases, the original image was first displayed and photographed as a control, then the image tape re-read in and the actual processing carried out, the reconstructed image outputted on the CRT and itself photographed.

* This is often referred to as non-uniform quantization DPCM or non-uniform step DPCM.

The measurement of the pel and pel difference frequency distributions as outlined in section 3 was the first item dealt with. This consisted in obtaining a frequency distribution of pel amplitudes for MOON, GIRL and COUPLE. (See Fig 3 - I).

Next the distribution of pel difference (Δ) amplitudes (where Δ_i is defined as

$$\Delta_i = pel_i - \rho \times pel_{i-1})$$

was obtained for different values of the correlation coefficient ρ. This was also done for all three image, the results appearing in figure 3 - II . Examination of these distributions clearly demonstrates the close similarity of the pel difference statistics for values of ρ close to unity.

A companding scheme applied to DPCM was then arbitrarily devised and tested; it appears in figure 4 - I . This scheme was then incorporated in the DPCM system as follows: a pel value was read in from the HSR, the difference between this and the previous pel was calculated, then this was inputted to the compression subroutine, which outputted a compressed difference. The compressed difference was then "transmitted" through a noiseless channel, the compressed difference expanded, and an estimate of the original pel reconstructed by addition to the previous reconstructed pel. (A block diagram of this system can be examined in figure 4 - II).

As expected, this scheme produced reconstructed images in which on the average, the magnitude of the differences between the original and reconstructed pels was found to be large

Range of DPCM coder outputs (octal, absolute value)	Compressor output* (octal, absolute value)	Expander output (octal, absolute valu
0	0	0
1	1	1
2	2	2
3, 4, 5	3	4
6, 7, 10	4	7
11, 12, 13, 14, 15	5	13
16, 17, 20, 21, 22	6	20
23, 24, 25, 26, 27, 30, 31	7	26
32, 33, 34, 35, 36, 37, 40	10	35
41, 42, 43, 44, 45, 46, 47, 50, 51	11	45
52, 53, 54, 55, 56, 57, 60, 61, 62	12	56
63, 64, 65, 66, 67, 70, 71, 72, 73, 74, 75	13	70
76, 77, 100, 101, ... 400	14	103

* Compressor output coded as 4-bit word, + sign bit, so transmitted as 5-bit word. Left-over words are $\pm 15_8$, $\pm 16_8$, $+ 17_8$. These can be used as special symbols, such as a TEST sequence, START SCAN, STOP SCAN, etc.

Fig. 4-I : Companding scheme used for DPCM coder output in all simulations.

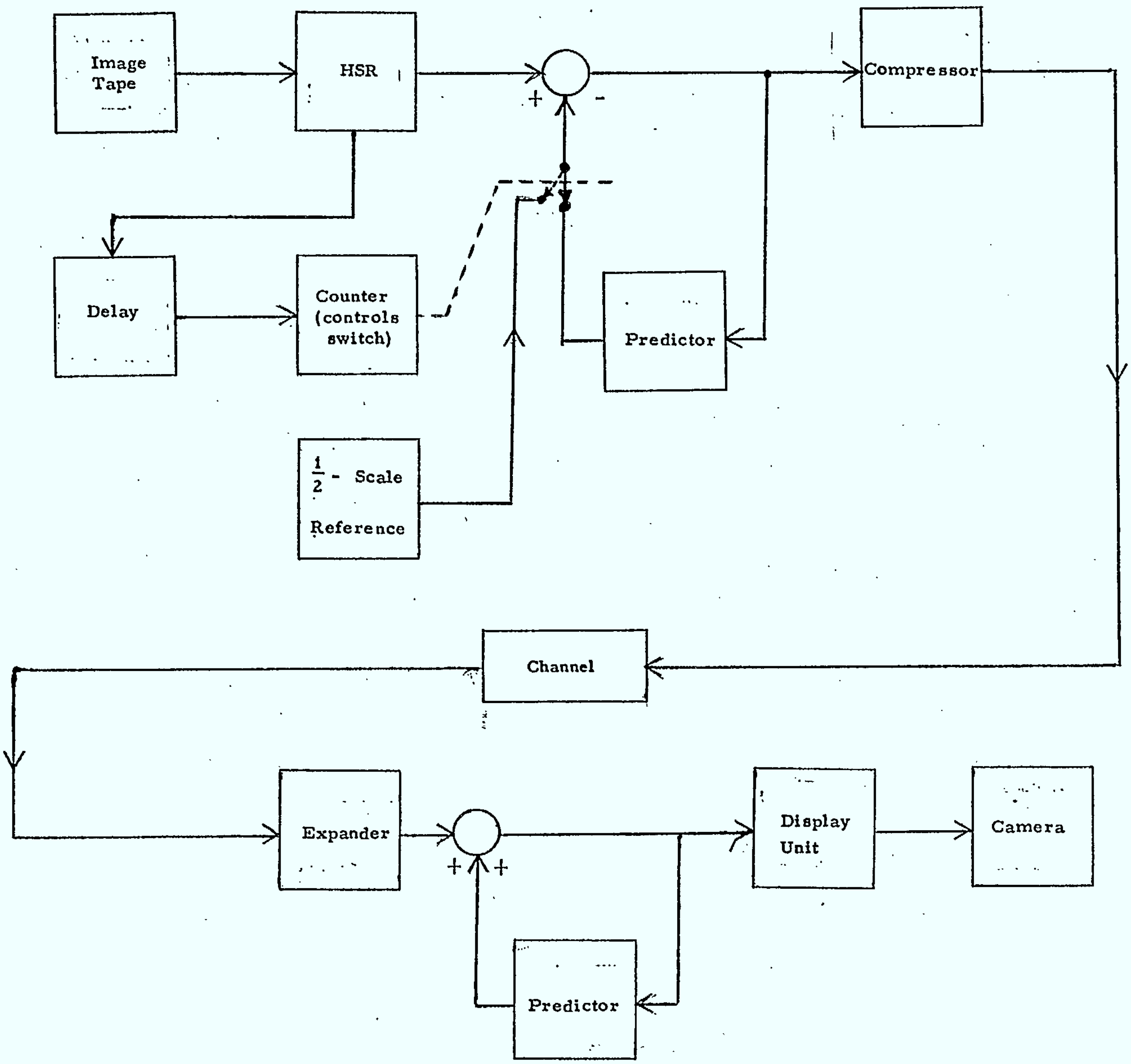


Fig. 4-II : Block diagram of image processing and transmission simulation system that does not incorporate a feedback loop in the XMTR. This was the original simulation system. The delay, counter and $\frac{1}{2}$ - scale reference blocks are used to generate the previous pel value for the first pel of each scan line.

and appear in bursts in the reconstructed image lines. The original image and their reconstructed counterparts can be examined in figure 4 -III . Since the scanning is from right to left and top to bottom, near the right-hand margins the reconstructed images are very good approximations to the original. However, the approximation deteriorates in bursts along a scanline as the left-hand margin is approached.

To improve the quality of the reconstructed image, the original companded DPCM system was modified by insertion of a feedback loop in the XMTR section. This feedback loop reconstructs the previous pel at the XMTR exactly as it will be reconstructed at the RCVR, given a noiseless channel, and so the difference between the previous reconstructed pel and the current pel can be compressed and "transmitted". The configuration of the RCVR itself remains unchanged. Figure 4 - IV is a block diagram of this system.

Using this modified companded DPCM simulation scheme, the image transmission simulations were repeated. The results appear in Fig. 4 - V . A subjective comparison of the reconstructed images in Figures 4 - III and 4 - V clearly reveals the improvement in image quality provided by the modified system.

Finally, to provide some immunity to channel noise, an algorithm to replace a pel difference by a PCM update value at every k pels is being developed. (This is required since the feedback loop at the XMTR is located entirely within the XMTR

(a)

(b)

Fig. 4-III: (a) Original 8-bit/pel PCM version of MOON and (b) 5-bit/pel reconstruction of MOON, using a companded DPCM system without feedback in the XMTR (Fig. 4-II).

(c)

(d)

Fig. 4-III:(c) Original 8-bit/pel PCM version of GIRL and (d) 5-bit/pel reconstruction of GIRL, using a companded DPCM system without feedback in the XMTR. (Fig. 4-II)

(e)

(f)

Fig. 4-III: (e) Original 8-bit/pel PCM version of COUPLE and (f) 5-bit/pel reconstruction of COUPLE, using a companded DPCM system without feedback in the XMTR. (Fig. 4-II).

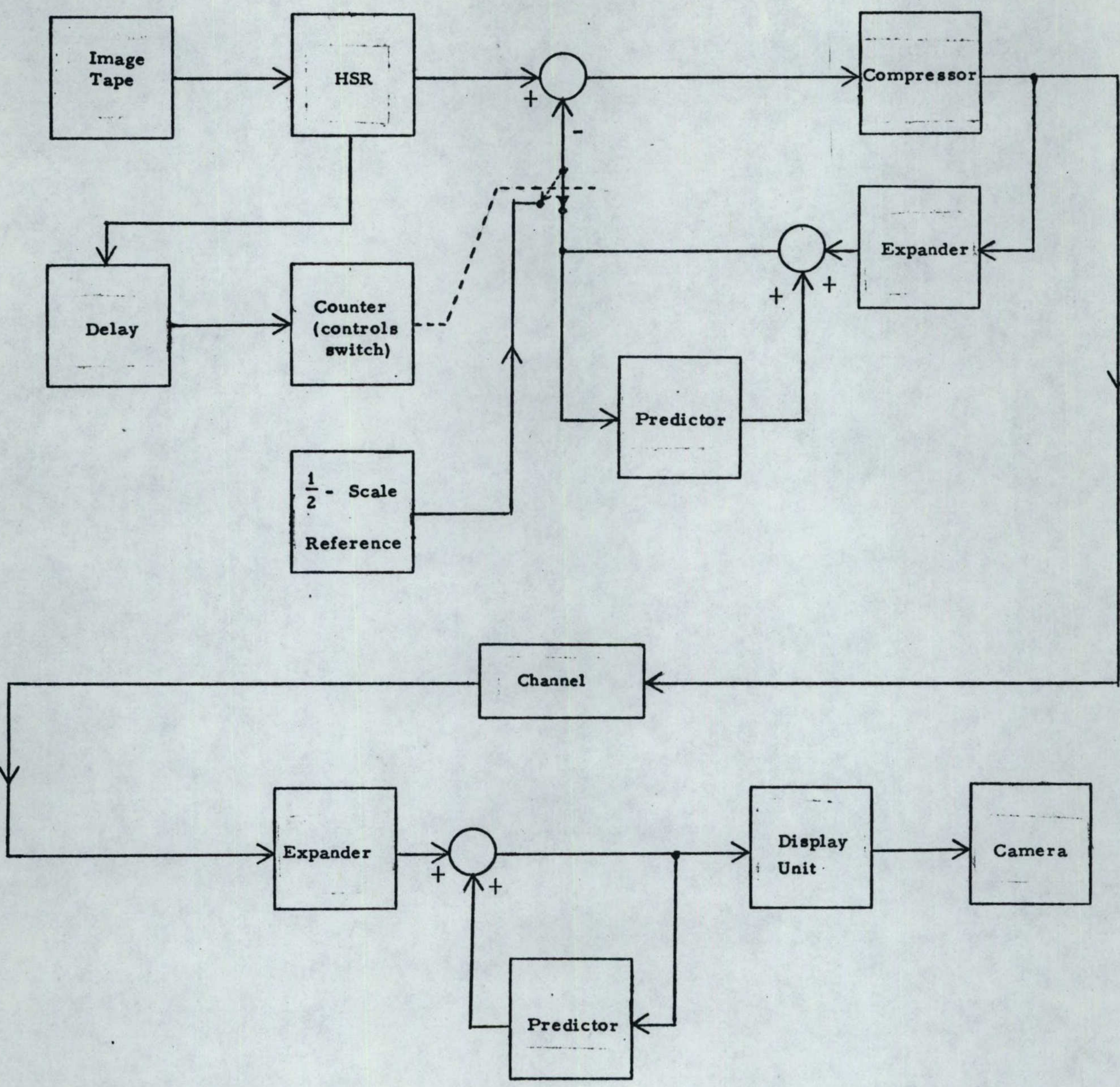
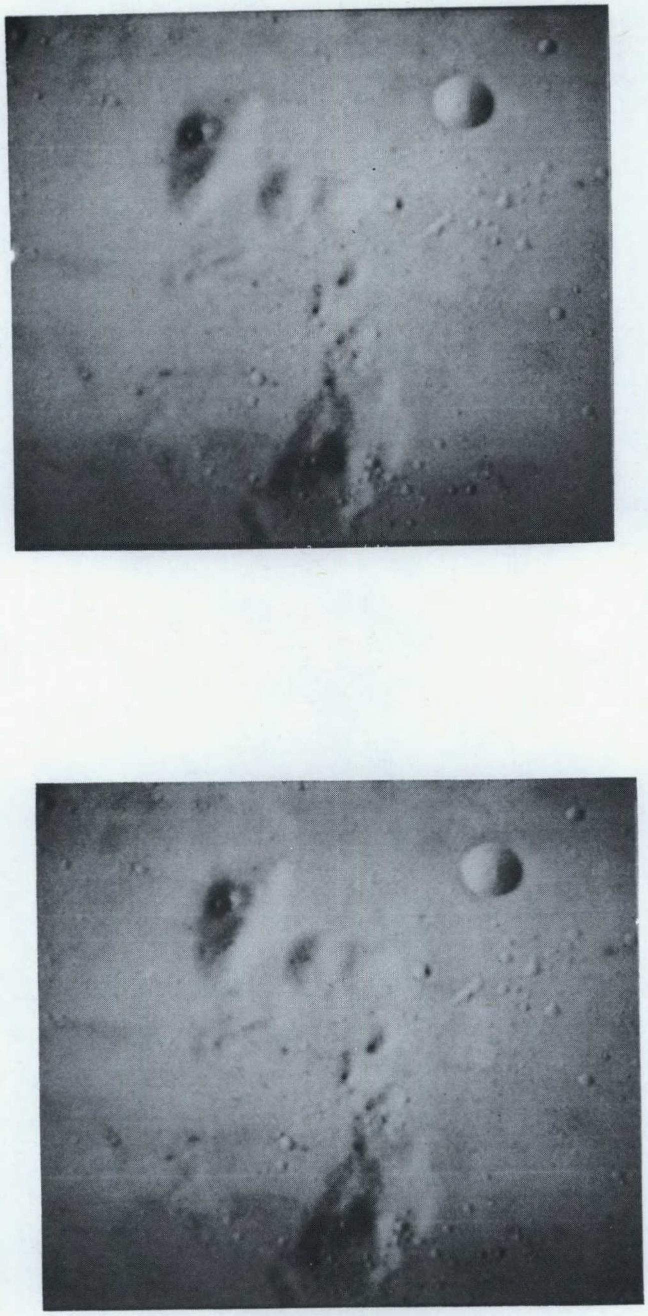


Fig. 4-IV: Block diagram of system used for digital image processing and transmission simulations at the University of Ottawa.



(a)

(b)

Fig. 4-V: (a) Original 8-bit/pel PCM version of MOON and (b) 5-bit/pel reconstruction of MOON, using a companded DPCM system with feedback in the XMTR. (Fig. 4-IV).

(c)

(d)

Fig. 4-V : (c) Original 8-bit/pel PCM version of GIRL and (d) 5-bit/pel reconstruction of GIRL, using a companded DPCM system with feedback in the XMTR (Fig. 4-IV).

(e)

(f)

Fig. 4-V : (e) Original 8-bit/pel PCM version of COUPLE and (f) 5-bit/pel reconstruction of COUPLE, using a companded DPCM system with feedback in the XMTR (Fig. 4-IV).

section and has no knowledge of channel errors. Hence whenever a channel error occurs, the reconstructed pel at the RCVR and the output of the XMTR feedback loop will not be the same, so that the feedback loop will not tend to reduce the effects of channel noise.) However, transmission of an 8-bit PCM word interleaved in a string of either 5 or 4-bit DPCM words requires more complex hardware to implement, and by the fact that the PCM word is twice as long as the DPCM word, it is twice as likely to be affected by channel noise. Also, an error occuring in a PCM updating word can cause a much larger error in the reconstructed image than a DPCM word, since the DPCM word will be companded and limited in amplitude by the expander characteristic, while the PCM updating word, with its much greater. dynamic range, would be transmitted directly. The use of a PCM update would also slightly decrease the BWC, by an amount proportional to the BER of the channel, since the number of updates used per picture would be proportional to the channel BER to achieve the same quality of reconstructed image.

*For these reasons, the updating word was chosen to be a DPCM word, its value being the difference between the half-scale grey level (200_8) and the pel value in the original image. Thus the update, while only as exact as the companding scheme can make it, will be very close to what the PCM value would be, while requiring no more bits than any other DPCM word and reducing the probability of error in the updating word by half, as compared to 8-bit PCM, since it is half the length. It also simplifies hardware requirements slightly, since all words would have the same length. Figure 4 - VI

* For reasons of simplicity, in the remainder of this discussion, a DPCM system utilizing 4-bit DPCM words will be assumed.

is a block diagram of a system which could be used to simulate the effects of channel noise on the system and the ability of the system to recover from channel errors.

In a practical system, it should be possible to transmit a test sequence before an image is transmitted, get an estimate on the current BER for the channel, then adjust the value of k accordingly. This would require some feedback from the RCVR. If 5-bit DPCM words were used, some of the "left-over" words (see Fig. 4 - I) could be used to directly transmit the value of k before transmission of the picture.

To simulate this latest system, an algorithm to produce errors at random locationsin the 2-D image must be implemented, then the error sequence added to the transmitted words, the number of errors per image being proportional to the BER. It seems conceptually possible to transmit reasonably high-quality images for relatively large error rates at little or no cost in BWC, the extra expenses occurring in the hardware. Of course, for maximum efficiency, the value of k should adapt itself to the changing BER of the channel. However, implementation of this type of system might conceivably increase the bit rate, since test sequences of some kind would probably have to be transmitted, say at every two lines of the image. The implementation would definitely be awkward, and probably not worth the expense, since the BER would not conceivably change very much while one image is being transmitted . For the case where a connection is made through a telephone voice channel, once the connection is made, the channel would probably not change appreciably until another connection was made.

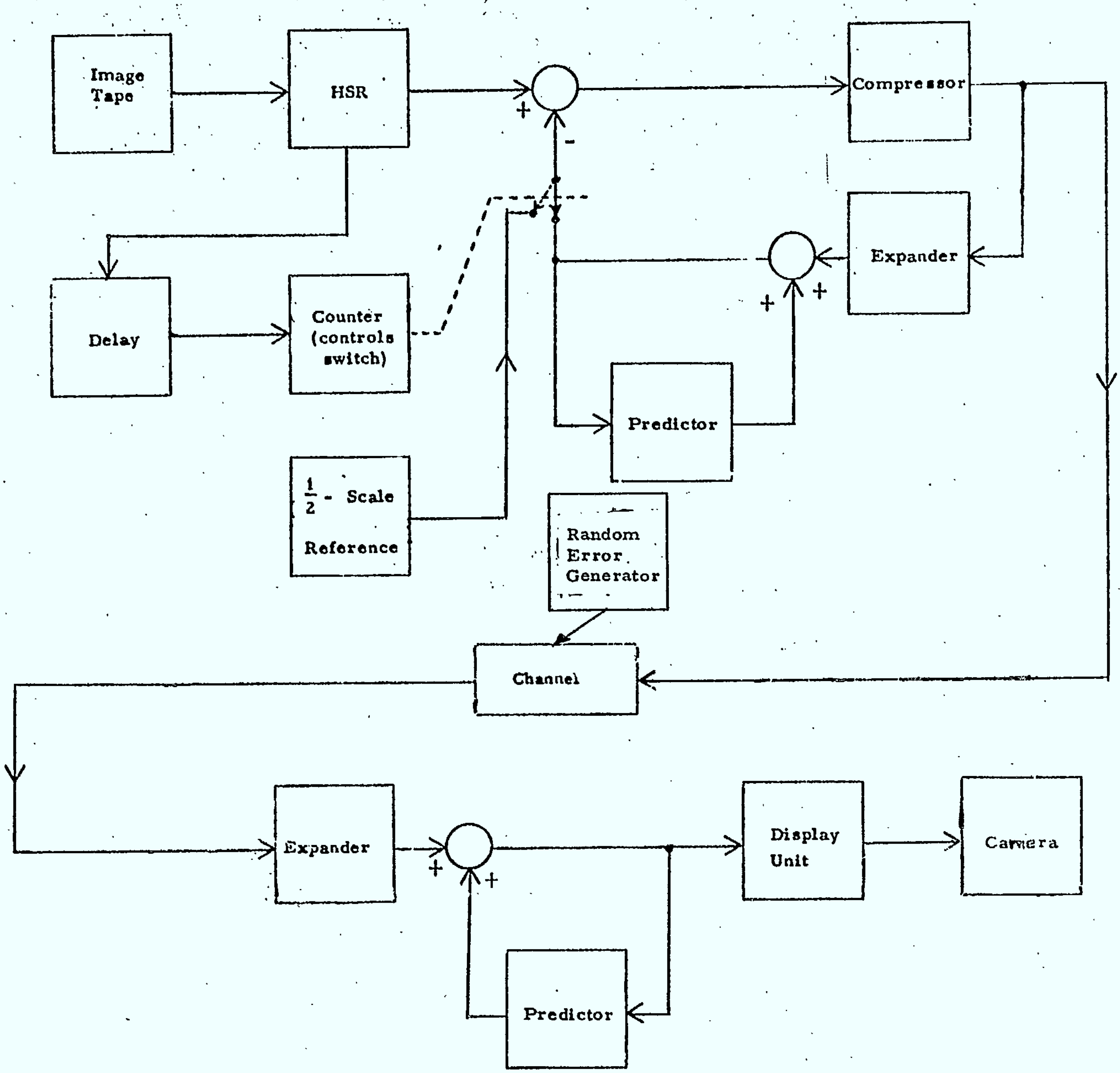


Fig. 4-VI: Block diagram of companded DPCM image processing and transmission system used for simulations, with added capability of simulating channel errors, for a given BER.

Clearly, there would be at most one connection made per image, so the implementation of a system where k was adaptive would probably be of very little help.

In all of the DPCM simulations, since the correlation between the last pel of say the i^{th} line of an image and the first pel of the $(i+1)^{th}$ line can be expected to be very low, the DPCM word transmitted for the 1st pel of any line is actually the difference between the actual pel value and half the grey scale range (being 200_8 in this case). Hence a large error does not occur in the first few pels of a line of the reconstructed image.

Another point which should be made clear at this time is the inherent synchronization of this system. Since the received and reconstructed images always have the same dimensions, and since all transmitted words are of the same length, framing should be perfect at the receiver as long as the framing circuit gates at the correct time. This can be ensured by transmitting a unique START word; the end can be determined at the receiver when the next pel would overflow the standard image dimensions.

Hence it should be possible to reconstruct high-quality images with a BWC of $\frac{8}{5}$: 1 with respect to 8-bit PCM, using the above system, even in the presence of a relatively high BER. However, the system is not optimized, and further work must be done to find, for example, the optimum compression characteristic. Optimized compression schemes have been developed using a mse criterion [17], but this criterion is not necessarily a valid one for image processing systems. Hence some further experimental work should give useful results.

A BWC of $\frac{8}{5}$: 1 is not very impressive, but as will be seen in section 5, the addition of some more hardware should provide a means of producing another large BWC factor multiplying that achieved with the DPCM system described above. The system comprising the DPCM system with added transformations will henceforth be referred to as the proposed system, to be described in the next section.

The BWC factor of $\frac{8}{5}$: 1 could in all probability be changed to a factor of $\frac{8}{4}$: 1 if the 5-bit DPCM is coded as 4-bit DPCM. The only disadvantage of this is the loss of the "unused" words (see fig. 4-I) and probably of some image quality as well. These unused words could be used as a START command for the DPCM receiver or for implementing changes, given adaptive processes. Otherwise these words could be used simply to represent DPCM levels, giving theoretically a slightly improved image quality, though it is doubtful that the change would be even noticed in a practical system. Clearly, there is room for improvement, even of the basic DPCM system as described above, but a BWC greater than 2:1 could probably not be achieved without a serious degradation in image quality.

5. Proposed System:

As discussed in the previous section, the system could obviously be improved upon. It might be possible to find a method of coding the symbol stream emerging from the DPCM coder before transmission in a very efficient manner such that BWC could again be achieved, giving a larger overall system BWC factor. One class of codes which have been suggested are the negacyclic codes [18] . They are nearly unique in their ability to operate upon + ve and - ve multi-level signals, instead of on binary data alone, and can be easily generated since they are quasi-cyclic.

The system would operate as follows: the negacyclic coder (NC) would examine the output of the DPCM coder, and when the sum of the absolute values of the emerging multi-level signal had reached a certain value, the NC would interpret this n-length sequence as being an error pattern, generate a syndrome of length s, and transmit the syndrome itself to the RCVR. At the RCVR, the s-length syndrome would be decoded to produce the n-length "error pattern", and this inputted to a DPCM decoder as in the previously mentioned system. If, on the average, $n >> s$, then a further BWC of $\left(\frac{n}{s}\right)$ will be achieved.

It may turn out that the type of output from the DPCM coder is not suitable for immediate negacyclic coding. In this case, intermediate processing on the DPCM output would transform it to a form more amenable to negacyclic coding. Fig. 5 - I is a block diagram of the proposed simulation system.

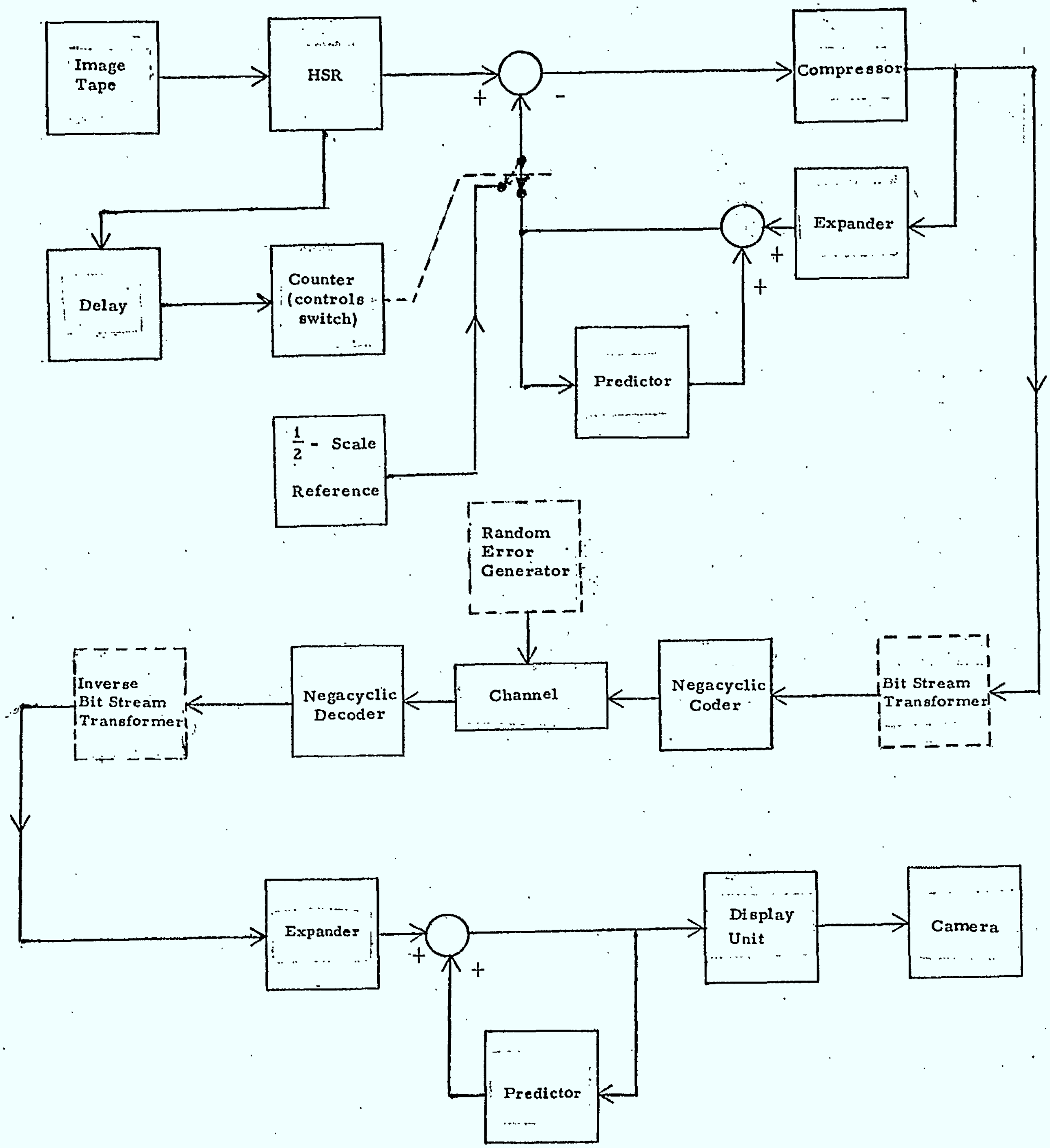


Fig. 5-I : Block diagram of proposed simulation system.

The system will complement one at the image processing lab of CRC at Shirley Bay. The two systems will be linked so that some of the conclusions arrived at on an analytical basis can be checked under real life conditions.

One practical aspect of the system that will be of great interest will undoubtedly be its sensitivity to channel noise. A little thought immediately shows that the updating scheme used in the system without negacyclic coding will be of no help here. If an error occurs in a transmitted syndrome, then of course it will be decoded incorrectly at the RCVR, thus producing an incorrect input to the RCVR DPCM decoder. Thus the outputs of the XMTR DPCM decoder and RCVR DPCM decoder will not be identical, and so the updating scheme will fail.

This problem may be alleviated provided an appropriate error - correcting code can be found such that the loss in BWC will be small compared to the channel noise immunity provided. This can be most simply studied by simulation, using a random number generating algorithm to produce errors in random locations in transmitted code words and viewing the effect on the reconstructed image for various BER.

6. Concluding Remarks

This report introduced the image transmission problem with some of the more important attempts which have been made to date to solve it. These have been briefly examined as to their relative merits and disadvantages.

Next, using the three standard digitized images, MOON, GIRL and COUPLE, a study of pel value as well as pel difference frequency distributions was made. It was concluded that in the second case the statistics were very similar , while the pel value frequency distributions were definitely not,being different for all three images.

Using these facts, companded DPCM followed by negacyclic coding was chosen as the "best" source coding scheme for the case under investigation. However, the system as proposed may still possibly turn out to require too complex an implementation and so be too expensive to have any practical applications.

A simulation of the companded DPCM portion of the system (see Fig. 4- II) was implemented on a PDP-8 mini-computer, and the results indicated that some control was required on the reconstruction error otherwise it could become relatively large. This error control was then implemented in the form of a feedback loop producing a reconstructed pel at the transmitter itself. Then a method was suggested to reduce the effects of channel noise on the reconstructed image and was found to be effective.

The BWC of slightly less than 2:1 produced by this system was judged to be too small to justify the system cost in hardware, so a negacyclic coder was inserted between the DPCM coder and the channel. This coder considers DPCM sequences as representing error patterns and outputs a short syndrome which uniquely describes the "error pattern". The DPCM portion of the RCVR is then preceded by a negacyclic decoder which will output DPCM sequences, given a syndrome. Larger overall bandwidth reduction factors could be expected to result from this system.

Clearly, while the results to date are promising, only preliminary work has been done. However a library of image processing and display routines has been developed to run on the PDP-8, so the simulation system exists in its basic form, and could easily be altered or added to by using appropriate sub-routines. Of course, the use of different input devices will necessitate additional A/D & D/A converters, interfacing hardware, etc.

Another point that must be stressed is that the final system configuration will not necessarily be as outlined in the report. For example it might be found worthwhile to use pre-and post-emphasis on the analog image signal before A/D and D/A conversions. If it is found that the approach taken is either not feasible for some reason(s), or gives unsatisfactory results, alternatives will be tried. Again since the requirements are for a physically-realizable system, little time will be spent on a system which may be excellent in theory but is found to be unworkable in practice, as these systems are only of academic interest.

The most interesting aspect of this proposal lies of course in the use of negacyclic codes to achieve BWC. The code itself can be simply generated using cyclic techniques, even though the code is not strictly cyclic ; the greater part of the work will consist in making a judicious choice of a proper negacyclic code. V.C. Chau [19] of the University of Ottawa, in a thesis to be published soon, has derived many generator polynomials for negacyclic codes of different lengths, and the choice of code will conveivably be made from these. Appendix C contains some background on negacyclic codes while Appendix D gives lists of Galois field elements for some of these codes as well as the corresponding generator polynomials.

Appendix A: Pel Distributions

Pel Distribution, MOON:*

Pel Value	No. of Occurences	Pel Value	No. of Occurences	Value	No. of Occurences
+0000	0076	+0045	0031	+0090	0206
+0001	0015	+0046	0042	+0091	0233
+0002	0009	+0047	0040	+0092	0226
+0003	0015	+0048	0040	+0093	0283
+0004	0013	+0049	0053	+0094	0283
+0005	0013	+0050	0052	+0095	0286
+0006	0015	+0051	0045	+0096	0279
+0007	0020	+0052	0050	+0097	0325
+0008	0001	+0053	0066	+0098	0371
+0009	0011	+0054	0054	+0099	0392
+0010	0014	+0055	0055	+0100	0394
+0011	0022	+0056	0078	+0101	0427
+0012	0018	+0057	0056	+0102	0474
+0013	0020	+0058	0068	+0103	0530
+0014	0019	+0059	0074	+0104	0531
+0015	0013	+0060	0071	+0105	0564
+0016	0017	+0061	0079	+0106	0558
+0017	0016	+0062	0063	+0107	0663
+0018	0022	+0063	0065	+0108	0675
+0019	0020	+0064	0069	+0109	0715
+0020	0018	+0065	0092	+0110	0698
+0021	0010	+0066	0102	+0111	0720
+0022	0018	+0067	0093	+0112	0726
+0023	0009	+0068	0100	+0113	0716
+0024	0012	+0069	0106	+0114	0797
+0025	0016	+0070	0088	+0115	0830
+0026	0016	+0071	0081	+0116	0851
+0027	0012	+0072	0068	+0117	0870
+0028	0007	+0073	0114	+0118	0924
+0029	0019	+0074	0097	+0119	0872
+0030	0021	+0075	0118	+0120	0862
+0031	0019	+0076	0111	+0121	0904
+0032	0022	+0077	0118	+0122	0881
+0033	0020	+0078	0100	+0123	0906
+0034	0027	+0079	0104	+0124	1005
+0035	0013	+0080	0116	+0125	0947
+0036	0019	+0081	0110	+0126	1102
+0037	0023	+0082	0117	+0127	1121
+0038	0022	+0083	0145	+0128	1034
+0039	0025	+0084	0150	+0129	1051
+0040	0025	+0085	0136	+0130	1030
+0041	0026	+0086	0179	+0131	1127
+0042	0033	+0087	0170	+0132	1118
+0043	0027	+0088	0184	+0133	1104
+0044	0037	+0089	0202	+0134	1054

* In all distributions, only non-zero occurences are indicated.

Pel Value	No. of Occurences	Pel Value	No. of Occurences	Pel Value	No. of Occurences
+0135	1117	+0180	0039	+0225	0003
+0136	1094	+0181	0032	+0226	0005
+0137	1301	+0182	0031	+0227	0001
+0138	1247	+0183	0028	+0228	0007
+0139	1099	+0184	0034	+0229	0002
+0140	1103	+0185	0029	+0231	0004
+0141	1074	+0186	0026	+0232	0002
+0142	1047	+0187	0022	+0233	0003
+0143	1016	+0188	0029	+0234	0003
+0144	1035	+0189	0025	+0235	0005
+0145	0934	+0190	0022	+0236	0007
+0146	1012	+0191	0015	+0237	0005
+0147	0948	+0192	0027	+0238	0003
+0148	1053	+0193	0019	+0239	0009
+0149	1091	+0194	0029	+0240	0012
+0150	0929	+0195	0022	+0241	0006
+0151	0904	+0196	0019	+0242	0006
+0152	0846	+0197	0017	+0243	0009
+0153	0802	+0198	0023	+0244	0006
+0154	0687	+0199	0018	+0245	0004
+0155	0778	+0200	0011	+0246	0003
+0156	0727	+0201	0016	+0247	0004
+0157	0729	+0202	0014	+0248	0001
+0158	0675	+0203	0028	+0249	0001
+0159	0587	+0204	0018		
+0160	0499	+0205	0019		
+0161	0451	+0206	0024		
+0162	0369	+0207	0013		
+0163	0364	+0208	0008		
+0164	0327	+0209	0011		
+0165	0340	+0210	0012		
+0166	0263	+0211	0019		
+0167	0249	+0212	0007		
+0168	0216	+0213	0009		
+0169	0172	+0214	0009		
+0170	0141	+0215	0011		
+0171	0110	+0216	0007		
+0172	0105	+0217	0005		
+0173	0093	+0218	0013		
+0174	0077	+0219	0005		
+0175	0061	+0220	0008		
+0176	0044	+0221	0003		
+0177	0058	+0222	0005		
+0178	0050	+0223	0003		
+0179	0048	+0224	0003		

Pel Distribution, GIRL:

Pel Value	No. of Occurences
+0000	0011
+0001	0096
+0002	0110
+0003	0105
+0004	0193
+0005	0181
+0006	0229
+0007	0165
+0008	0162
+0009	0217
+0010	0152
+0011	0177
+0012	0205
+0013	0241
+0014	0264
+0015	0331
+0016	0198
+0017	0383
+0018	0186
+0019	0428
+0020	0207
+0021	0205
+0022	0500
+0023	0277
+0024	0304
+0025	0200
+0026	0349
+0027	0441
+0028	0416
+0029	0467
+0030	0502
+0031	0511
+0032	0457
+0033	0509
+0034	0465
+0035	0597
+0037	0567
+0038	0838
+0039	0876
+0040	1049
+0042	1119
+0043	1335
+0044	1437
+0046	1691
+0047	1663

Pel Value	No. of Occurences
+0049	1924
+0050	1779
+0052	1708
+0053	1434
+0055	1458
+0057	1210
+0058	1050
+0060	0791
+0062	0778
+0064	0706
+0065	0765
+0067	0726
+0069	0929
+0071	0814
+0073	0922
+0075	0890
+0077	0945
+0079	0971
+0081	1005
+0083	0940
+0086	1032
+0088	0933
+0090	0955
+0092	0845
+0095	0893
+0097	0862
+0099	0871
+0102	0882
+0104	0925
+0107	0905
+0109	0941
+0112	0842
+0115	0864
+0117	0885
+0120	0917
+0123	0751
+0125	0796
+0128	0687
+0131	0719
+0134	0627
+0136	0587
+0139	0468
+0142	0452
+0145	0310
+0148	0350

Pel Value	No. of Occuren-ces
+0151	0287
+0154	0274
+0157	0202
+0160	0220
+0164	0196
+0167	0187
+0170	0158
+0173	0138
+0176	0162
+0180	0138
+0183	0132
+0186	0115
+0189	0116
+0193	0137
+0197	0108
+0200	0118
+0203	0122
+0207	0109
+0210	0097
+0214	0112
+0218	0095
+0221	0082
+0225	0050
+0229	0041
+0232	0033
+0236	0022
+0240	0012
+0244	0003
+0247	0003
+0251	0004
+0254	0003

Pel Distribution, COUPLE:

Pel Value	No. of Occurences	Pel Value	No. of Occurences	Pel Value	No. of Occurences
+0000	0007	+0049	0857	+0148	0151
+0001	3794	+0050	0849	+0151	0144
+0002	1549	+0052	0852	+0154	0108
+0003	2407	+0053	0798	+0157	0078
+0004	2814	+0055	0924	+0160	0064
+0005	1329	+0057	0833	+0164	0034
+0006	1114	+0058	0883	+0167	0021
+0007	0791	+0060	0848	+0170	0017
+0008	0763	+0062	0948	+0173	0010
+0009	0772	+0064	0907	+0176	0011
+0010	0645	+0065	0956	+0180	0011
+0011	0643	+0067	0823	+0183	0008
+0012	0722	+0069	0980	+0186	0011
+0013	0665	+0071	0766	+0189	0005
+0014	0684	+0073	0760	+0193	0007
+0015	0807	+0075	0732	+0197	0003
+0016	0463	+0077	0699	+0200	0004
+0017	1268	+0079	0650	+0203	0004
+0018	0706	+0081	0632	+0207	0002
+0019	1675	+0083	0532	+0210	0002
+0020	0788	+0086	0590	+0214	0001
+0021	0759	+0088	0544	+0225	0003
+0022	1485	+0090	0505	+0229	0002
+0023	0678	+0092	0492	+0236	0001
+0024	0621	+0094	0001	+0240	0001
+0025	0363	+0095	0478		
+0026	0597	+0097	0456		
+0027	0622	+0099	0443		
+0028	0554	+0102	0369		
+0029	0640	+0104	0370		
+0030	0556	+0107	0352		
+0031	0572	+0109	0339		
+0032	0638	+0112	0285		
+0033	0727	+0115	0257		
+0034	0666	+0117	0218		
+0035	0737	+0120	0240		
+0037	0681	+0123	0164		
+0038	0799	+0125	0167		
+0039	0672	+0128	0128		
+0040	0691	+0131	0134		
+0042	0741	+0134	0128		
+0043	0779	+0136	0105		
+0044	0803	+0139	0095		
+0046	0884	+0142	0127		
+0047	0834	+0145	0120		

Appendix B: Difference Distributions

Pel Difference Distribution, MOON; $\rho = 1.00$.

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
-0149	0001	-0055	0011	-0010	1249
-0139	0001	-0054	0018	-0009	1510
-0118	0001	-0053	0012	-0008	1726
-0113	0001	-0052	0010	-0007	1964
-0103	0001	-0051	0012	-0006	2207
-0102	0001	-0050	0011	-0005	2419
-0101	0001	-0049	0019	-0004	2552
-0099	0001	-0048	0022	-0003	2822
-0094	0002	-0047	0018	-0002	2899
-0092	0001	-0046	0018	-0001	3044
-0091	0002	-0045	0029	+0000	3245
-0089	0002	-0044	0033	+0001	2895
-0088	0002	-0043	0027	+0002	2966
-0087	0002	-0042	0028	+0003	2837
-0086	0001	-0041	0030	+0004	2584
-0085	0001	-0040	0034	+0005	2342
-0084	0002	-0039	0043	+0006	2011
-0083	0003	-0038	0046	+0007	1866
-0082	0001	-0037	0061	+0008	1696
-0081	0001	-0036	0056	+0009	1491
-0080	0001	-0035	0060	+0010	1256
-0079	0002	-0034	0072	+0011	1056
-0078	0002	-0033	0081	+0012	0946
-0077	0002	-0032	0077	+0013	0819
-0076	0002	-0031	0078	+0014	0674
-0075	0001	-0030	0101	+0015	0570
-0074	0003	-0029	0113	+0016	0516
-0073	0003	-0028	0130	+0017	0474
-0072	0002	-0027	0131	+0018	0374
-0071	0006	-0026	0163	+0019	0354
-0070	0003	-0025	0165	+0020	0288
-0069	0004	-0024	0181	+0021	0306
-0068	0003	-0023	0214	+0022	0220
-0067	0006	-0022	0243	+0023	0201
-0066	0004	-0021	0285	+0024	0173
-0065	0010	-0020	0325	+0025	0164
-0064	0008	-0019	0311	+0026	0162
-0063	0005	-0018	0407	+0027	0134
-0062	0007	-0017	0476	+0028	0123
-0061	0008	-0016	0535	+0029	0116
-0060	0006	-0015	0594	+0030	0100
-0059	0013	-0014	0724	+0031	0098
-0058	0009	-0013	0848	+0032	0092
-0057	0017	-0012	0957	+0033	0073
-0056	0011	-0011	1147	+0034	0072

Pel Difference Values	No. of Occurences
+0035	0059
+0036	0055
+0037	0057
+0038	0046
+0039	0041
+0040	0034
+0041	0032
+0042	0033
+0043	0032
+0044	0023
+0045	0027
+0046	0014
+0047	0017
+0048	0018
+0049	0011
+0050	0020
+0051	0016
+0052	0016
+0053	0015
+0054	0012
+0055	0011
+0056	0009
+0057	0008
+0058	0008
+0059	0007
+0060	0005
+0061	0006
+0062	0006
+0063	0005
+0064	0008
+0065	0004
+0066	0004
+0067	0003
+0068	0004
+0069	0012
+0070	0002
+0071	0002
+0072	0004
+0073	0004
+0075	0001
+0076	0001
+0078	0004
+0079	0002
+0080	0001
+0081	0001
+0083	0001
+0084	0001
+0085	0002
+0087	0001
+0089	0001
+0090	0002
+0091	0002
+0093	0003
+0097	0001
+0099	0001
+0101	0001
+0103	0001
+0104	0001
+0105	0001
+0106	0001
+0109	0002
+0110	0001
+0111	0001
+0114	0001
+0120	0001
+0137	0001
+0142	0001

Pel Difference Distribution, MOON; $\rho = 0.99$

Pel Difference Values	No. of Occurences	Pel Differences Values	No. of Occurences	Pel Difference Values	No. of Occurences
-0148	0001	-0049	0007	-0004	2405
-0116	0001	-0048	0012	-0003	2534
-0112	0001	-0047	0016	-0002	2744
-0101	0002	-0046	0025	-0001	2904
-0099	0001	-0045	0013	+0000	3159
-0098	0001	-0044	0026	+0001	3126
-0093	0001	-0043	0024	+0002	2957
-0092	0001	-0042	0026	+0003	2955
-0090	0002	-0041	0031	+0004	2926
-0089	0001	-0040	0024	+0005	2648
-0087	0001	-0039	0032	+0006	2434
-0086	0003	-0038	0041	+0007	2089
-0085	0002	-0037	0038	+0008	1890
-0083	0002	-0036	0050	+0009	1751
-0082	0001	-0035	0060	+0010	1558
-0081	0002	-0034	0048	+0011	1306
-0080	0002	-0033	0064	+0012	1098
-0078	0001	-0032	0066	+0013	0968
-0077	0003	-0031	0077	+0014	0835
-0075	0002	-0030	0070	+0015	0690
-0074	0002	-0029	0088	+0016	0571
-0073	0002	-0028	0111	+0017	0516
-0072	0002	-0027	0105	+0018	0482
-0071	0003	-0026	0125	+0019	0390
-0070	0003	-0025	0145	+0020	0355
-0069	0003	-0024	0174	+0021	0288
-0068	0002	-0023	0168	+0022	0309
-0067	0002	-0022	0195	+0023	0229
-0066	0001	-0021	0233	+0024	0200
-0065	0004	-0020	0271	+0025	0182
-0064	0007	-0019	0326	+0026	0165
-0063	0013	-0018	0327	+0027	0162
-0062	0001	-0017	0389	+0028	0142
-0061	0002	-0016	0466	+0029	0123
-0060	0003	-0015	0560	+0030	0112
-0059	0007	-0014	0583	+0031	0101
-0058	0007	-0013	0676	+0032	0090
-0057	0010	-0012	0805	+0033	0099
-0056	0009	-0011	0983	+0034	0072
-0055	0008	-0010	1101	+0035	0076
-0054	0015	-0009	1222	+0036	0055
-0053	0009	-0008	1463	+0037	0058
-0052	0013	-0007	1673	+0038	0058
-0051	0013	-0006	1946	+0039	0047
-0050	0008	-0005	2146	+0040	0042

Pel Differences Values	No. of Occurences
+0041	0027
+0042	0039
+0043	0031
+0044	0033
+0045	0022
+0046	0026
+0047	0015
+0048	0023
+0049	0015
+0050	0013
+0051	0016
+0052	0015
+0053	0017
+0054	0015
+0055	0013
+0056	0010
+0057	0009
+0058	0007
+0059	0008
+0060	0007
+0061	0006
+0062	0006
+0063	0005
+0064	0005
+0065	0009
+0066	0002
+0067	0005
+0068	0003
+0069	0005
+0070	0011
+0071	0002
+0072	0002
+0073	0004
+0074	0004
+0076	0001
+0077	0001
+0079	0004
+0080	0002
+0081	0001
+0082	0001
+0084	0001
+0085	0001
+0086	0002
+0088	0001
+0090	0001

Pel Differences Values	No. of Occurences
+0091	0003
+0094	0003
+0097	0001
+0100	0001
+0102	0001
+0104	0001
+0106	0002
+0109	0001
+0110	0001
+0111	0001
+0112	0001
+0115	0001
+0121	0001
+0143	0001

Pel Difference Distributions, MOON; $\rho = 0.98$

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
-0146	0001	-0048	0008	-0003	2314
-0114	0001	-0047	0010	-0002	2523
-0111	0001	-0046	0015	-0001	2656
-0108	0001	-0045	0020	+0000	2889
-0100	0001	-0044	0018	+0001	3011
-0098	0001	-0043	0021	+0002	3126
-0097	0002	-0042	0026	+0003	3041
-0095	0001	-0041	0027	+0004	2924
-0091	0002	-0040	0028	+0005	2979
-0089	0001	-0039	0021	+0006	2771
-0088	0002	-0038	0031	+0007	2545
-0086	0001	-0037	0042	+0008	2278
-0085	0002	-0036	0034	+0009	1939
-0084	0003	-0035	0045	+0010	1833
-0083	0001	-0034	0061	+0011	1613
-0082	0002	-0033	0049	+0012	1431
-0080	0002	-0032	0069	+0013	1196
-0079	0003	-0031	0066	+0014	1042
-0078	0001	-0030	0058	+0015	0881
-0076	0002	-0029	0080	+0016	0751
-0075	0002	-0028	0089	+0017	0607
-0074	0001	-0027	0098	+0018	0541
-0072	0002	-0026	0121	+0019	0499
-0071	0003	-0025	0113	+0020	0433
-0070	0003	-0024	0135	+0021	0352
-0069	0003	-0023	0158	+0022	0320
-0068	0002	-0022	0173	+0023	0297
-0067	0002	-0021	0183	+0024	0278
-0066	0003	-0020	0238	+0025	0214
-0065	0001	-0019	0244	+0026	0179
-0064	0004	-0018	0298	+0027	0175
-0063	0006	-0017	0324	+0028	0182
-0062	0008	-0016	0359	+0029	0129
-0061	0006	-0015	0425	+0030	0138
-0060	0003	-0014	0493	+0031	0111
-0058	0007	-0013	0567	+0032	0108
-0057	0004	-0012	0630	+0033	0101
-0056	0008	-0011	0759	+0034	0086
-0055	0012	-0010	0905	+0035	0081
-0054	0007	-0009	0999	+0036	0074
-0053	0009	-0008	1197	+0037	0058
-0052	0013	-0007	1357	+0038	0063
-0051	0011	-0006	1572	+0039	0055
-0050	0013	-0005	1865	+0040	0048
-0049	0012	-0004	1974	+0041	0049

Pel Difference Values	No. of Occurences
+0042	0032
+0043	0038
+0044	0032
+0045	0030
+0046	0023
+0047	0023
+0048	0019
+0049	0024
+0050	0014
+0051	0012
+0052	0012
+0053	0021
+0054	0018
+0055	0013
+0056	0012
+0057	0006
+0058	0012
+0059	0005
+0060	0010
+0061	0006
+0062	0005
+0063	0006
+0064	0005
+0065	0007
+0066	0007
+0067	0003
+0068	0006
+0069	0003
+0070	0004
+0071	0010
+0072	0003
+0073	0003
+0074	0004
+0075	0003
+0077	0002
+0079	0001
+0080	0005
+0081	0001
+0082	0001
+0085	0001
+0086	0002
+0087	0001
+0089	0001
+0091	0002
+0092	0003

Pel Difference Values	No. of Occurences
+0094	0001
+0095	0002
+0101	0001
+0104	0002
+0105	0001
+0106	0001
+0107	0001
+0110	0002
+0112	0002
+0116	0001
+0121	0001
+0143	0001

Pel Difference Distribution, GIRL; $\rho = 1.00$

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
-0127	0001	-0076	0015	-0030	0074
-0126	0001	-0075	0012	-0029	0047
-0119	0001	-0074	0002	-0028	0068
-0118	0002	-0073	0003	-0027	0056
-0117	0005	-0072	0002	-0026	0085
-0116	0009	-0071	0004	-0025	0109
-0115	0010	-0070	0004	-0024	0113
-0114	0007	-0069	0004	-0023	0118
-0113	0009	-0067	0006	-0022	0161
-0112	0006	-0066	0005	-0021	0164
-0111	0018	-0065	0004	-0020	0152
-0110	0008	-0064	0001	-0019	0237
-0109	0020	-0063	0004	-0018	0183
-0108	0010	-0062	0003	-0017	0261
-0107	0002	-0061	0002	-0016	0339
-0106	0014	-0060	0002	-0015	0328
-0105	0007	-0059	0005	-0014	0469
-0104	0009	-0058	0005	-0013	0565
-0103	0007	-0057	0008	-0012	0525
-0102	0011	-0056	0006	-0011	0858
-0101	0007	-0055	0005	-0010	0797
-0100	0009	-0054	0005	-0009	1054
-0099	0006	-0053	0011	-0008	1475
-0098	0008	-0052	0009	-0007	1490
-0097	0003	-0051	0005	-0006	1936
-0096	0006	-0050	0011	-0005	2671
-0095	0004	-0049	0008	-0004	2370
-0094	0004	-0048	0009	-0003	4022
-0093	0004	-0047	0008	-0002	4337
-0092	0002	-0046	0010	-0001	2559
-0091	0005	-0045	0013	+0000	8116
-0090	0005	-0044	0013	+0001	2702
-0089	0010	-0043	0006	+0002	4363
-0088	0027	-0042	0018	+0003	3923
-0087	0002	-0041	0016	+0004	2393
-0086	0039	-0040	0016	+0005	2575
-0085	0049	-0039	0021	+0006	1947
-0084	0031	-0038	0027	+0007	1426
-0083	0004	-0037	0033	+0008	1453
-0082	0052	-0036	0020	+0009	1032
-0081	0041	-0035	0026	+0010	0818
-0080	0005	-0034	0038	+0011	0940
-0079	0044	-0033	0028	+0012	0558
-0078	0018	-0032	0041	+0013	0593
-0077	0006	-0031	0042	+0014	0510

Pel Difference Values	No. of Occurences
+ØØ15	Ø358
+ØØ16	Ø42Ø
+ØØ17	Ø3Ø1
+ØØ18	Ø235
+ØØ19	Ø257
+ØØ2Ø	Ø165
+ØØ21	Ø2Ø4
+ØØ22	Ø173
+ØØ23	Ø123
+ØØ24	Ø113
+ØØ25	Ø118
+ØØ26	Ø112
+ØØ27	ØØ68
+ØØ28	ØØ71
+ØØ29	ØØ53
+ØØ3Ø	ØØ73
+ØØ31	ØØ57
+ØØ32	ØØ44
+ØØ33	ØØ47
+ØØ34	ØØ51
+ØØ35	ØØ41
+ØØ36	ØØ25
+ØØ37	ØØ36
+ØØ38	ØØ36
+ØØ39	ØØ25
+ØØ4Ø	ØØ27
+ØØ41	ØØ16
+ØØ42	ØØ26
+ØØ43	ØØ14
+ØØ44	ØØ26
+ØØ45	ØØ13
+ØØ46	ØØ12
+ØØ47	ØØ2Ø
+ØØ48	ØØØ9
+ØØ49	ØØØ8
+ØØ5Ø	ØØ12
+ØØ51	ØØØ4
+ØØ52	ØØØ9
+ØØ53	ØØØ9
+ØØ54	ØØØ3
+ØØ55	ØØØ3
+ØØ56	ØØØ3
+ØØ57	ØØØ6
+ØØ58	ØØØ8
+ØØ59	ØØØ6

Pel Difference Values	No. of Occurences
+ØØ6Ø	ØØØ5
+ØØ61	ØØØ3
+ØØ62	ØØØ5
+ØØ63	ØØØ3
+ØØ64	ØØØ2
+ØØ65	ØØØ2
+ØØ66	ØØØ2
+ØØ68	ØØØ1
+ØØ7Ø	ØØØ2
+ØØ71	ØØØ2
+ØØ72	ØØØ2
+ØØ73	ØØØ1
+ØØ74	ØØØ2
+ØØ76	ØØØ1
+ØØ79	ØØØ1
+ØØ82	ØØØ1
+ØØ89	ØØØ1
+ØØ9Ø	ØØØ1
+ØØ93	ØØØ1
+ØØ95	ØØØ1
+Ø112	ØØØ1

Pel Difference Distribution, GIRL; $\rho = 0.99$

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
-0126	0007	-0080	0042	-0035	0026
-0125	0009	-0079	0004	-0034	0024
-0124	0010	-0078	0045	-0033	0038
-0123	0006	-0077	0020	-0032	0034
-0122	0001	-0076	0006	-0031	0028
-0120	0003	-0075	0018	-0030	0060
-0119	0001	-0074	0016	-0029	0066
-0118	0007	-0073	0001	-0028	0046
-0117	0001	-0072	0005	-0027	0072
-0116	0001	-0071	0002	-0026	0063
-0115	0003	-0070	0005	-0025	0080
-0114	0002	-0069	0003	-0024	0113
-0113	0003	-0068	0003	-0023	0100
-0112	0005	-0067	0006	-0022	0120
-0111	0001	-0066	0004	-0021	0145
-0110	0005	-0065	0005	-0020	0184
-0109	0004	-0064	0003	-0019	0144
-0108	0001	-0063	0005	-0018	0235
-0107	0003	-0062	0003	-0017	0200
-0106	0001	-0061	0003	-0016	0254
-0105	0003	-0060	0004	-0015	0346
-0104	0003	-0059	0002	-0014	0295
-0103	0002	-0058	0004	-0013	0508
-0102	0002	-0057	0002	-0012	0572
-0101	0002	-0056	0008	-0011	0552
-0100	0003	-0055	0007	-0010	0898
-0099	0001	-0054	0008	-0009	0813
-0098	0003	-0053	0005	-0008	1063
-0097	0005	-0052	0009	-0007	1662
-0096	0003	-0051	0007	-0006	1509
-0095	0002	-0050	0010	-0005	2207
-0094	0005	-0049	0009	-0004	3006
-0093	0007	-0048	0011	-0003	2784
-0092	0003	-0047	0009	-0002	8271
-0091	0002	-0046	0002	-0001	4805
-0090	0004	-0045	0016	+0000	3796
-0089	0009	-0044	0011	+0001	6903
-0088	0011	-0043	0012	+0002	2573
-0087	0031	-0042	0009	+0003	4400
-0086	0003	-0041	0017	+0004	3089
-0085	0040	-0040	0014	+0005	2146
-0084	0056	-0039	0020	+0006	2619
-0083	0038	-0038	0016	+0007	1470
-0082	0005	-0037	0031	+0008	1358
-0081	0054	-0036	0031	+0009	1488

Pel Difference Values	No. of Occurences
+0010	0942
+0011	0730
+0012	0917
+0013	0621
+0014	0494
+0015	0557
+0016	0353
+0017	0345
+0018	0357
+0019	0217
+0020	0250
+0021	0179
+0022	0202
+0023	0154
+0024	0136
+0025	0112
+0026	0114
+0027	0107
+0028	0060
+0029	0087
+0030	0044
+0031	0078
+0032	0056
+0033	0038
+0034	0051
+0035	0048
+0036	0037
+0037	0030
+0038	0037
+0039	0029
+0040	0028
+0041	0028
+0042	0014
+0043	0026
+0044	0014
+0045	0022
+0046	0016
+0047	0012
+0048	0019
+0049	0007
+0050	0009
+0051	0011
+0052	0005
+0053	0011
+0054	0005
+0055	0005
+0056	0002
+0057	0004
+0058	0007
+0059	0008
+0060	0004
+0061	0004
+0062	0003
+0063	0006
+0064	0003
+0065	0002
+0066	0002
+0067	0001
+0068	0001
+0070	0001
+0071	0001
+0072	0003
+0073	0001
+0074	0001
+0075	0002
+0076	0001
+0079	0001
+0082	0001
+0090	0001
+0091	0001
+0094	0001
+0096	0001
+0113	0001

Pel Difference Distribution, GIRL; $\rho = 0.98$

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
-0123	0002	-0078	0041	-0033	0029
-0122	0004	-0077	0004	-0032	0028
-0121	0007	-0076	0044	-0031	0041
-0120	0004	-0075	0021	-0030	0029
-0119	0006	-0074	0002	-0029	0052
-0118	0005	-0073	0019	-0028	0054
-0117	0004	-0072	0011	-0027	0050
-0116	0003	-0071	0003	-0026	0079
-0115	0002	-0070	0002	-0025	0057
-0114	0004	-0069	0003	-0024	0100
-0113	0002	-0068	0004	-0023	0086
-0112	0005	-0067	0002	-0022	0101
-0111	0005	-0066	0003	-0021	0140
-0110	0010	-0065	0004	-0020	0112
-0109	0003	-0064	0006	-0019	0198
-0108	0001	-0063	0001	-0018	0134
-0107	0001	-0062	0003	-0017	0195
-0106	0005	-0061	0006	-0016	0261
-0105	0002	-0060	0004	-0015	0234
-0104	0002	-0059	0004	-0014	0328
-0103	0005	-0058	0002	-0013	0391
-0102	0002	-0057	0002	-0012	0408
-0101	0004	-0056	0006	-0011	0582
-0100	0003	-0055	0003	-0010	0656
-0099	0009	-0054	0007	-0009	0729
-0098	0002	-0053	0011	-0008	1111
-0097	0009	-0052	0006	-0007	1021
-0096	0005	-0051	0008	-0006	1703
-0095	0004	-0050	0009	-0005	1926
-0094	0003	-0049	0010	-0004	2094
-0093	0005	-0048	0008	-0003	3115
-0092	0006	-0047	0012	-0002	3704
-0091	0001	-0046	0006	-0001	3816
-0090	0009	-0045	0007	+0000	4889
-0089	0003	-0044	0013	+0001	5047
-0088	0006	-0043	0011	+0002	4659
-0087	0009	-0042	0011	+0003	3686
-0086	0013	-0041	0009	+0004	3768
-0085	0028	-0040	0016	+0005	2537
-0084	0002	-0039	0014	+0006	2516
-0083	0040	-0038	0016	+0007	2267
-0082	0052	-0037	0023	+0008	1300
-0081	0031	-0036	0030	+0009	1650
-0080	0005	-0035	0031	+0010	1119
-0079	0055	-0034	0024	+0011	0988

Pel Difference Values	No. of Occurences
+0012	0823
+0013	0825
+0014	0535
+0015	0667
+0016	0461
+0017	0314
+0018	0442
+0019	0260
+0020	0279
+0021	0212
+0022	0209
+0023	0181
+0024	0141
+0025	0157
+0026	0105
+0027	0112
+0028	0078
+0029	0101
+0030	0055
+0031	0066
+0032	0078
+0033	0044
+0034	0056
+0035	0035
+0036	0053
+0037	0025
+0038	0036
+0039	0042
+0040	0022
+0041	0029
+0042	0021
+0043	0021
+0044	0024
+0045	0019
+0046	0017
+0047	0013
+0048	0020
+0049	0010
+0050	0012
+0051	0006
+0052	0012
+0053	0008
+0054	0006
+0055	0006
+0056	0003

Pel Difference Values	No. of Occurences
+0057	0001
+0058	0006
+0059	0007
+0060	0006
+0061	0003
+0062	0006
+0063	0004
+0064	0006
+0065	0002
+0066	0002
+0067	0001
+0068	0001
+0069	0001
+0071	0002
+0073	0003
+0075	0003
+0076	0002
+0079	0001
+0083	0001
+0091	0002
+0095	0001
+0097	0001
+0114	0001

Pel Difference Distribution , COUPLE ; $\rho = 1.00$

Pel Difference Values	No. of Occurences
-0174	0001
-0155	0001
-0154	0001
-0152	0002
-0145	0001
-0144	0001
-0142	0002
-0139	0001
-0137	0001
-0135	0001
-0131	0001
-0130	0001
-0129	0002
-0127	0035
-0126	0027
-0125	0068
-0124	0022
-0123	0001
-0121	0003
-0119	0001
-0118	0003
-0117	0002
-0116	0002
-0115	0001
-0114	0004
-0113	0006
-0112	0002
-0111	0011
-0110	0013
-0109	0039
-0108	0023
-0107	0041
-0106	0084
-0105	0043
-0104	0026
-0103	0008
-0102	0014
-0101	0009
-0100	0006
-0099	0006
-0098	0002
-0097	0004
-0096	0004
-0095	0004
-0094	0002

Pel Difference Values	No. of Occurences
-0093	0003
-0091	0002
-0090	0002
-0088	0004
-0087	0003
-0086	0003
-0085	0004
-0084	0004
-0083	0001
-0082	0003
-0081	0005
-0080	0002
-0079	0006
-0078	0003
-0077	0003
-0076	0006
-0075	0005
-0074	0007
-0073	0011
-0072	0005
-0071	0010
-0070	0011
-0069	0004
-0068	0008
-0067	0009
-0066	0014
-0065	0011
-0064	0015
-0063	0011
-0062	0015
-0061	0012
-0060	0015
-0059	0018
-0058	0019
-0057	0013
-0056	0012
-0055	0028
-0054	0014
-0053	0020
-0052	0026
-0051	0019
-0050	0017
-0049	0024
-0048	0017
-0047	0014
-0046	0021

Pel Difference Values	No. of Occurences
-0045	0019
-0044	0018
-0043	0013
-0042	0031
-0041	0019
-0040	0034
-0039	0036
-0038	0027
-0037	0042
-0036	0035
-0035	0046
-0034	0041
-0033	0050
-0032	0048
-0031	0052
-0030	0065
-0029	0061
-0028	0082
-0027	0079
-0026	0089
-0025	0110
-0024	0107
-0023	0119
-0022	0086
-0021	0158
-0020	0148
-0019	0198
-0018	0222
-0017	0241
-0016	0313
-0015	0317
-0014	0357
-0013	0451
-0012	0465
-0011	0610
-0010	0663
-0009	0873
-0008	1072
-0007	1168
-0006	1476
-0005	1811
-0004	2078
-0003	2587
-0002	4118
-0001	5034

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
+0000	13052	+0045	0031	+0104	0001
+0001	4694	+0046	0017	+0105	0001
+0002	4049	+0047	0020	+0116	0001
+0003	2682	+0048	0018	+0117	0001
+0004	2109	+0049	0024	+0121	0001
+0005	1754	+0050	0026	+0123	0001
+0006	1412	+0051	0020	+0127	0002
+0007	1163	+0052	0036	+0135	0001
+0008	0987	+0053	0018	+0138	0001
+0009	0816	+0054	0015	+0146	0001
+0010	0629	+0055	0016		
+0011	0625	+0056	0014		
+0012	0470	+0057	0010		
+0013	0398	+0058	0018		
+0014	0329	+0059	0014		
+0015	0296	+0060	0017		
+0016	0284	+0061	0007		
+0017	0218	+0062	0013		
+0018	0232	+0063	0012		
+0019	0181	+0064	0011		
+0020	0169	+0065	0012		
+0021	0187	+0066	0004		
+0022	0164	+0067	0003		
+0023	0143	+0068	0005		
+0024	0133	+0069	0004		
+0025	0127	+0070	0009		
+0026	0109	+0071	0002		
+0027	0128	+0072	0001		
+0028	0087	+0073	0004		
+0029	0067	+0074	0005		
+0030	0079	+0075	0002		
+0031	0058	+0076	0001		
+0032	0067	+0078	0001		
+0033	0068	+0079	0003		
+0034	0048	+0080	0001		
+0035	0056	+0081	0002		
+0036	0038	+0082	0003		
+0037	0051	+0084	0001		
+0038	0044	+0086	0004		
+0039	0038	+0087	0001		
+0040	0050	+0089	0002		
+0041	0025	+0090	0001		
+0042	0044	+0091	0001		
+0043	0027	+0099	0001		
+0044	0026	+0103	0002		

Pel Difference Distribution, COUPLE; $\rho = 0.99$

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
-0172	0001	-0092	0004	-0046	0010
-0153	0001	-0090	0005	-0045	0021
-0152	0001	-0089	0008	-0044	0022
-0150	0002	-0088	0003	-0043	0021
-0143	0001	-0087	0004	-0042	0015
-0142	0001	-0086	0004	-0041	0027
-0140	0002	-0085	0005	-0040	0023
-0137	0001	-0084	0003	-0039	0039
-0135	0001	-0083	0006	-0038	0033
-0133	0001	-0082	0001	-0037	0026
-0129	0001	-0081	0003	-0036	0043
-0128	0001	-0080	0005	-0035	0037
-0127	0002	-0079	0003	-0034	0049
-0126	0035	-0078	0005	-0033	0038
-0125	0045	-0077	0006	-0032	0052
-0124	0060	-0076	0003	-0031	0048
-0123	0009	-0075	0004	-0030	0053
-0122	0001	-0074	0005	-0029	0068
-0121	0001	-0073	0006	-0028	0057
-0118	0003	-0072	0011	-0027	0091
-0117	0001	-0071	0005	-0026	0084
-0116	0003	-0070	0010	-0025	0103
-0115	0002	-0069	0013	-0024	0108
-0114	0002	-0068	0004	-0023	0111
-0113	0001	-0067	0007	-0022	0112
-0112	0003	-0066	0010	-0021	0106
-0111	0002	-0065	0012	-0020	0159
-0110	0004	-0064	0011	-0019	0152
-0109	0007	-0063	0017	-0018	0213
-0108	0014	-0062	0011	-0017	0227
-0107	0018	-0061	0014	-0016	0266
-0106	0010	-0060	0016	-0015	0314
-0105	0033	-0059	0012	-0014	0323
-0104	0014	-0058	0022	-0013	0402
-0103	0020	-0057	0019	-0012	0462
-0102	0011	-0056	0015	-0011	0500
-0101	0024	-0055	0010	-0010	0688
-0100	0018	-0054	0030	-0009	0714
-0099	0016	-0053	0017	-0008	0959
-0098	0023	-0052	0022	-0007	1188
-0097	0022	-0051	0022	-0006	1308
-0096	0023	-0050	0023	-0005	1658
-0095	0023	-0049	0015	-0004	2091
-0094	0018	-0048	0031	-0003	2535
-0093	0014	-0047	0017	-0002	3556

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
-0001	6315	+0044	0017	+0103	0002
+0000	8985	+0045	0036	+0105	0002
+0001	6832	+0046	0020	+0117	0001
+0002	2821	+0047	0019	+0118	0001
+0003	3384	+0048	0022	+0121	0001
+0004	2082	+0049	0015	+0123	0001
+0005	1771	+0050	0027	+0127	0001
+0006	1634	+0051	0025	+0128	0001
+0007	1177	+0052	0027	+0135	0001
+0008	1091	+0053	0029	+0138	0001
+0009	0844	+0054	0012	+0147	0001
+0010	0737	+0055	0017		
+0011	0596	+0056	0015		
+0012	0540	+0057	0013		
+0013	0454	+0058	0014		
+0014	0340	+0059	0012		
+0015	0340	+0060	0020		
+0016	0282	+0061	0011		
+0017	0240	+0062	0002		
+0018	0262	+0063	0018		
+0019	0172	+0064	0014		
+0020	0203	+0065	0010		
+0021	0168	+0066	0007		
+0022	0178	+0067	0003		
+0023	0142	+0068	0004		
+0024	0140	+0069	0006		
+0025	0134	+0070	0003		
+0026	0105	+0071	0008		
+0027	0131	+0072	0002		
+0028	0097	+0073	0001		
+0029	0080	+0074	0005		
+0030	0064	+0075	0003		
+0031	0078	+0076	0003		
+0032	0060	+0079	0003		
+0033	0064	+0080	0002		
+0034	0062	+0082	0004		
+0035	0047	+0083	0001		
+0036	0046	+0084	0001		
+0037	0048	+0086	0002		
+0038	0043	+0087	0002		
+0039	0038	+0088	0001		
+0040	0041	+0089	0001		
+0041	0048	+0090	0002		
+0042	0024	+0092	0001		
+0043	0049	+0099	0001		

Pel Difference Distribution, COUPLE; $\rho = 0.98$

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
-0170	0001	-0093	0003	-0045	0019
-0151	0001	-0092	0002	-0044	0017
-0150	0001	-0091	0003	-0043	0016
-0149	0001	-0090	0001	-0042	0020
-0148	0001	-0088	0002	-0041	0018
-0141	0001	-0087	0004	-0040	0029
-0140	0001	-0085	0001	-0039	0025
-0139	0002	-0084	0004	-0038	0039
-0136	0001	-0083	0004	-0037	0036
-0134	0001	-0082	0004	-0036	0016
-0132	0001	-0080	0002	-0035	0051
-0127	0002	-0079	0008	-0034	0036
-0126	0001	-0078	0006	-0033	0050
-0125	0001	-0077	0004	-0032	0045
-0124	0037	-0076	0005	-0031	0049
-0123	0026	-0075	0004	-0030	0048
-0122	0048	-0074	0003	-0029	0057
-0121	0028	-0073	0005	-0028	0066
-0120	0006	-0072	0008	-0027	0064
-0119	0004	-0071	0010	-0026	0103
-0118	0002	-0070	0006	-0025	0062
-0117	0002	-0069	0005	-0024	0116
-0116	0003	-0068	0011	-0023	0108
-0115	0002	-0067	0008	-0022	0112
-0114	0004	-0066	0006	-0021	0099
-0113	0003	-0065	0013	-0020	0133
-0112	0005	-0064	0010	-0019	0157
-0111	0007	-0063	0014	-0018	0169
-0110	0029	-0062	0013	-0017	0229
-0109	0018	-0061	0009	-0016	0231
-0108	0032	-0060	0022	-0015	0268
-0107	0016	-0059	0011	-0014	0332
-0106	0073	-0058	0017	-0013	0372
-0105	0042	-0057	0019	-0012	0375
-0104	0050	-0056	0015	-0011	0494
-0103	0037	-0055	0017	-0010	0576
-0102	0004	-0054	0009	-0009	0685
-0101	0005	-0053	0027	-0008	0889
-0100	0001	-0052	0016	-0007	0976
-0099	0002	-0051	0027	-0006	1326
-0098	0009	-0050	0020	-0005	1478
-0097	0003	-0049	0017	-0004	1855
-0096	0003	-0048	0020	-0003	2509
-0095	0001	-0047	0027	-0002	3446
-0094	0001	-0046	0012	-0001	5863

Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences	Pel Difference Values	No. of Occurences
+0000	11364	+0045	0035	+0103	0001
+0001	6162	+0046	0020	+0104	0001
+0002	3759	+0047	0022	+0105	0001
+0003	2949	+0048	0025	+0106	0001
+0004	2482	+0049	0013	+0118	0002
+0005	1968	+0050	0023	+0122	0001
+0006	1671	+0051	0023	+0124	0001
+0007	1501	+0052	0031	+0128	0002
+0008	1068	+0053	0031	+0136	0001
+0009	1090	+0054	0018	+0139	0001
+0010	0696	+0055	0014	+0148	0001
+0011	0723	+0056	0013		
+0012	0501	+0057	0015		
+0013	0566	+0058	0012		
+0014	0355	+0059	0015		
+0015	0366	+0060	0017		
+0016	0319	+0061	0013		
+0017	0228	+0062	0009		
+0018	0293	+0063	0008		
+0019	0193	+0064	0017		
+0020	0205	+0065	0011		
+0021	0161	+0066	0009		
+0022	0198	+0067	0006		
+0023	0144	+0068	0004		
+0024	0151	+0069	0004		
+0025	0136	+0070	0006		
+0026	0120	+0071	0006		
+0027	0134	+0072	0004		
+0028	0101	+0074	0004		
+0029	0105	+0075	0005		
+0030	0061	+0076	0001		
+0031	0078	+0077	0001		
+0032	0063	+0078	0001		
+0033	0063	+0079	0001		
+0034	0070	+0080	0003		
+0035	0052	+0082	0003		
+0036	0053	+0083	0002		
+0037	0034	+0084	0002		
+0038	0050	+0086	0001		
+0039	0050	+0087	0003		
+0040	0032	+0089	0002		
+0041	0053	+0091	0001		
+0042	0031	+0092	0001		
+0043	0041	+0093	0001		
+0044	0022	+0099	0001		

Appendix C : Negacyclic Codes

Appendix C: Negacyclic Codes [18]

A negacyclic (NC) code V of block length n over GF(p) (p prime > 2 and n a nonmultiple of p) is the set of all multiples of a generator polynomial g(X) which divides $X^n + 1$ over GF(p). The quotient $h(X) = \frac{X^n + 1}{g(X)}$ is called the parity check polynomial.

In the discussion of NC codes we require the Lee metric rather than the Hamming metric.

The Lee weight of the n-tuple $a_0\, a_1\, a_2 \ldots a_{n-1}$ over GF(p) is the sum of the Lee weights of a_i.

The Lee weight of $a_i = |a_i| = \pm\, a_i$ modulo p and $0 \leq |a_i| \leq \frac{p-1}{2}$.

For example, the sequence 1 2 -2 -1 0 -1 over GF(5) was the Lee weight $= |1| + |2| + |-2| + |-1| + |0| + |-1| = 7$.

By a t-random-error-correcting NC code V, we mean that V can correct any combination of t or fewer errors as long as the Lee weight of the error pattern is t or less.

For example, if t = 2 and p = 5, then the error polynomial E(X) can be $a_i X^i$ with $|a_i| = 0, 1, 2$; $a_i X^i + a_j X^j$ with $|a_i| + |a_j| \leq 2$.

It is of interest to note that, since in the correction of burst errors it is the length of the burst that is involved and not the weight of the burst, NC codes do not have any advantage over cyclic codes in regard to burst-error correction.

Expressing $V(X)$, an element of V, in the form

$$V(X) = a_0 + a_1 X + a_2 X^2 + \ldots + a_{n-1} X^{n-1},$$

let us consider

$$V'(X) = -a_{n-1} + a_0 X + a_1 X^2 + \ldots + a_{n-2} X^{n-1},$$

where $V'(X)$ is said to be an NC shift of $V(X)$. We note that

$$V'(X) = X\,V(X) - a_{n-1} X^n - a_{n-1}$$

or

$$V'(X) = X\,V(X) - a_{n-1} (X^n + 1).$$

Since $g(X)$ divides $V(X)$ and X^n+1, it follows that $g(X)$ divides $V'(X)$. Thus we can make the following

Comment: For every word of V, the NC shift of the word also belongs to B.

This indicates why NC codes are called what they are. The analogy between NC codes and cyclic codes is obvious.

Analogous to the class of cyclic BCH codes we have the known following

Theorem : If the roots of $g(X)$ of V include $\alpha, \alpha^3, \alpha^5, \ldots \alpha^{2t-1}$, where $2t - 1 < p$, then V is t-random-error-correcting.

From the theorem it is clear that the main restriction is that, for a given p, t has to be less than p+1. We recall that there is no such restriction in the case of BCH codes.

Since $g(X)$ divides $X^n + 1$, this also means that $g(X)$ divides $X^{2n} - 1$. Therefore $2n = p^r - 1$ or $n = \frac{p^r - 1}{2}$.
In view of this we have the following

Comment: With reference to the theorem, V has $n = \frac{p^r - 1}{2}$ and α is the primitive element of $GF(p^r)$.

For example, if $p = 5$ and $r = 2$ and $t = 2$, then we have $GF(p^r) = GF(2^5)$, $n = \frac{p^r - 1}{2} = 12$. If we construct $GF(2^5)$ using $X^2 + X + 2$ which is a primitive polynomial with degree $r = 2$, then we find that $\alpha^3 = \alpha^{2t-1}$ is a root of $X^2 - 2$. Therefore $g(X) = (X^2 + X + 2)\ (X^2 - 2) = X^4 + 2X^3 + X + 1$. Thus this $g(X)$ generates a (12,8) 2-error-correcting NC code.

Since $g(X)$ divides $X^n + 1$, where $n = \frac{p^r - 1}{2}$, the knowledge of the irreducible factors of $X^n + 1$ is vital to the design of NC codes.

<u>Negacyclic Codes With $n \neq \frac{p^r - 1}{2}$.</u>

The NC codes discussed previously have all lengths $n = \frac{p^r - 1}{2}$. Now we discuss the NC codes with lengths $n \neq \frac{p^r - 1}{2}$.

To find $g(X)$ for such codes we proceed as follows: for a given n and a given p, let u be the smallest odd positive integer such that $nu = \frac{p^r - 1}{2}$. Then $g(X)$ can be found from the following

<u>Theorem:</u> If the roots of $g(X)$ l include $\alpha^u, \alpha^{3u}, \ldots \alpha^{(2t-1)u}$, where $2t - 1 < p$, then the NC code of length n can correct any error pattern of Lee weight t or less and α is a primitive element of $GF(p^r)$.

This theorem follows directly from the theorem given previously for NC codes with length $n = \frac{p^r - 1}{2}$. The restriction that u is odd is necessary since if u is not odd then $X^n + 1$ will not divide $X^n + 1$.

Example : For p = 5 and r = 2, we have $v = \frac{p^r - 1}{2} = 12$. If n = 4, then $4 \cdot 3 = 12$ so that u = 3. The polynomial $-2 + X^2$ has $\alpha^u = \alpha^3$ as a root. Therefore, if we make $g(X) = -2 + X^2$, then this g(X) generates a (4, 2) NC code with t = 1.

We conclude this appendix with a list of generator polynomials for NC codes of certain lengths.

Appendix D : Some Negacyclic Code Generator Polynomials.

NEGACYCLIC CODES, LEE MFTRIC, P = 5,R = 2, N = 12

X** 12 + 1 =	ORDER OF ROOT
(-2 +(-2)X + X**2)	24
(-2 +(0)X + X**2)	8
(-2 +(2)X + X**2)	24
(2 +(-1)X + X**2)	24
(2 +(0)X + X**2)	8
(2 +(1)X + X**2)	24

TABLE OF ALL NON-ZERO ELEMENTS OVER GF(7**2) WITH PRIMITIVE POLYNOMIAL

P(X)=X**2 +(-2)*X + (-2)

```
A** 0 =           1
A** 1 =  1A
A** 2 =  2A +     2
A** 3 =  1A +     4
A** 4 =  1A +     2
A** 5 =  4A +     2
A** 6 =  0A +     3
A** 7 =  3A +     0
A** 8 =  1A +     1
A** 9 =  3A +     2
A** 10 = 3A +     1
A** 11 = 2A +     1
A** 12 = 0A +     4
A** 13 = 4A +     0
A** 14 = 3A +     3
A** 15 = 4A +     1
A** 16 = 4A +     3
A** 17 = 1A +     3
A** 18 = 0A +     2
A** 19 = 2A +     0
A** 20 = 4A +     4
A** 21 = 2A +     3
A** 22 = 2A +     4
A** 23 = 3A +     4
A** 24 = 0A +     1
```

GENERATOR POLYNOMIAL IS :

G(X) = X**4 +(-2)X**3 +(0)X**2 +(1)X +(1)

NEGACYCLIC CODES, LEE METRIC, P = 5,R = 3 ,N = 62

```
X** 62 + 1 =                                                 ORDER OF ROOT
          ( -2 +( -1)X +( -2)X**2 + X**3 )                        124
          ( -2 +(  2)X +( -2)X**2 + X**3 )                        124
          ( -2 +(  0)X +( -1)X**2 + X**3 )                        124
          ( -2 +( -2)X +(  0)X**2 + X**3 )                        124
          ( -2 +( -1)X +(  0)X**2 + X**3 )                        124
          ( -2 +( -1)X +(  1)X**2 + X**3 )                        124
          ( -2 +(  1)X +(  1)X**2 + X**3 )                        124
          ( -2 +(  0)X +(  2)X**2 + X**3 )                        124
          ( -2 +(  1)X +(  2)X**2 + X**3 )                        124
          ( -2 +(  2)X +(  2)X**2 + X**3 )                        124
          (  2 +(  0)X +( -2)X**2 + X**3 )                        124
          (  2 +(  1)X +( -2)X**2 + X**3 )                        124
          (  2 +(  2)X +( -2)X**2 + X**3 )                        124
          (  2 +( -1)X +( -1)X**2 + X**3 )                        124
          (  2 +(  1)X +( -1)X**2 + X**3 )                        124
          (  2 +( -2)X +(  0)X**2 + X**3 )                        124
          (  2 +( -1)X +(  0)X**2 + X**3 )                        124
          (  2 +(  0)X +(  1)X**2 + X**3 )                        124
          (  2 +( -1)X +(  2)X**2 + X**3 )                        124
          (  2 +(  2)X +(  2)X**2 + X**3 )                        124

                              ( RE(X) )

                           WHERE RE(X) =

                                           1*X**  0

                                           0*X**  1

                                           1*X**  2
```

TABLE OF ALL NON-ZERO ELEMENTS OVER GF(5**3) WITH PRIMITIVE POLYNOMIAL:

P(X)=X**3 +(-1)*X**2 + (Ø)*X + (-2)

A** 0 =						1
A** 1 =			1A			
A** 2 =	1A**2					
A** 3 =	1A**2	+	ØA	+		2
A** 4 =	1A**2	+	2A	+		2
A** 5 =	3A**2	+	2A	+		2
A** 6 =	ØA**2	+	2A	+		1
A** 7 =	2A**2	+	1A	+		Ø
A** 8 =	3A**2	+	ØA	+		4
A** 9 =	3A**2	+	4A	+		1
A** 10 =	2A**2	+	1A	+		1
A** 11 =	3A**2	+	1A	+		4
A** 12 =	4A**2	+	4A	+		1
A** 13 =	3A**2	+	1A	+		3
A** 14 =	4A**2	+	3A	+		1
A** 15 =	2A**2	+	1A	+		3
A** 16 =	3A**2	+	3A	+		4
A** 17 =	1A**2	+	4A	+		1
A** 18 =	ØA**2	+	1A	+		2
A** 19 =	1A**2	+	2A	+		Ø
A** 20 =	3A**2	+	ØA	+		2
A** 21 =	3A**2	+	2A	+		1
A** 22 =	ØA**2	+	1A	+		1
A** 23 =	1A**2	+	1A	+		Ø
A** 24 =	2A**2	+	ØA	+		2
A** 25 =	2A**2	+	2A	+		4
A** 26 =	4A**2	+	4A	+		4

A**	27	=	3A**2	+	4A	+	3
A**	28	=	2A**2	+	3A	+	1
A**	29	=	0A**2	+	1A	+	4
A**	30	=	1A**2	+	4A	+	0
A**	31	=	0A**2	+	0A	+	2
A**	32	=	0A**2	+	2A	+	0
A**	33	=	2A**2	+	0A	+	0
A**	34	=	2A**2	+	0A	+	4
A**	35	=	2A**2	+	4A	+	4
A**	36	=	1A**2	+	4A	+	4
A**	37	=	0A**2	+	4A	+	2
A**	38	=	4A**2	+	2A	+	0
A**	39	=	1A**2	+	0A	+	3
A**	40	=	1A**2	+	3A	+	2
A**	41	=	4A**2	+	2A	+	2
A**	42	=	1A**2	+	2A	+	3
A**	43	=	3A**2	+	3A	+	2
A**	44	=	1A**2	+	2A	+	1
A**	45	=	3A**2	+	1A	+	2
A**	46	=	4A**2	+	2A	+	1
A**	47	=	1A**2	+	1A	+	3
A**	48	=	2A**2	+	3A	+	2
A**	49	=	0A**2	+	2A	+	4
A**	50	=	2A**2	+	4A	+	0
A**	51	=	1A**2	+	0A	+	4
A**	52	=	1A**2	+	4A	+	2

A** 53 =	ØA**2 +	2A +	2
A** 54 =	2A**2 +	2A +	Ø
A** 55 =	4A**2 +	ØA +	4
A** 56 =	4A**2 +	4A +	3
A** 57 =	3A**2 +	3A +	3
A** 58 =	1A**2 +	3A +	1
A** 59 =	4A**2 +	1A +	2
A** 6Ø =	ØA**2 +	2A +	3
A** 61 =	2A**2 +	3A +	Ø
A** 62 =	ØA**2 +	ØA +	4
A** 63 =	ØA**2 +	4A +	Ø
A** 64 =	4A**2 +	ØA +	Ø
A** 65 =	4A**2 +	ØA +	3
A** 66 =	4A**2 +	3A +	3
A** 67 =	2A**2 +	3A +	3
A** 68 =	ØA**2 +	3A +	4
A** 69 =	3A**2 +	4A +	Ø
A** 7Ø =	2A**2 +	ØA +	1
A** 71 =	2A**2 +	1A +	4
A** 72 =	3A**2 +	4A +	4
A** 73 =	2A**2 +	4A +	1
A** 74 =	1A**2 +	1A +	4
A** 75 =	2A**2 +	4A +	2
A** 76 =	1A**2 +	2A +	4
A** 77 =	3A**2 +	4A +	2

```
A**  78 =  2A**2 +  2A +  1
A**  79 =  4A**2 +  1A +  4
A**  80 =  0A**2 +  4A +  3
A**  81 =  4A**2 +  3A +  0
A**  82 =  2A**2 +  0A +  3
A**  83 =  2A**2 +  3A +  4
A**  84 =  0A**2 +  4A +  4
A**  85 =  4A**2 +  4A +  0
A**  86 =  3A**2 +  0A +  3
A**  87 =  3A**2 +  3A +  1
A**  88 =  1A**2 +  1A +  1
A**  89 =  2A**2 +  1A +  2
A**  90 =  3A**2 +  2A +  4
A**  91 =  0A**2 +  4A +  1
A**  92 =  4A**2 +  1A +  0
A**  93 =  0A**2 +  0A +  3
A**  94 =  0A**2 +  3A +  0
A**  95 =  3A**2 +  0A +  0
A**  96 =  3A**2 +  0A +  1
A**  97 =  3A**2 +  1A +  1
A**  98 =  4A**2 +  1A +  1
A**  99 =  0A**2 +  1A +  3
A** 100 =  1A**2 +  3A +  0
A** 101 =  4A**2 +  0A +  2
A** 102 =  4A**2 +  2A +  3
A** 103 =  1A**2 +  3A +  3
```

```
A** 104 =   4A**2 +   3A +   2
A** 105 =   2A**2 +   2A +   3
A** 106 =   4A**2 +   3A +   4
A** 107 =   2A**2 +   4A +   3
A** 108 =   1A**2 +   3A +   4
A** 109 =   4A**2 +   4A +   2
A** 110 =   3A**2 +   2A +   3
A** 111 =   0A**2 +   3A +   1
A** 112 =   3A**2 +   1A +   0
A** 113 =   4A**2 +   0A +   1
A** 114 =   4A**2 +   1A +   3
A** 115 =   0A**2 +   3A +   3
A** 116 =   3A**2 +   3A +   0
A** 117 =   1A**2 +   0A +   1
A** 118 =   1A**2 +   1A +   2
A** 119 =   2A**2 +   2A +   2
A** 120 =   4A**2 +   2A +   4
A** 121 =   1A**2 +   4A +   3
A** 122 =   0A**2 +   3A +   2
A** 123 =   3A**2 +   2A +   0
A** 124 =   0A**2 +   0A +   1
```

```
GENERATOR POLYNOMIAL IS :
```

```
G(X) = X**6 +(  2)X**5 +( -1)X**4 +( -2)X**3 +( -3)X**2 +(  1)X +(  1
```

NEGACYCLIC CODES, LEE METRIC, P = 7,R = 2 , N = 24

X** 24 + 1 =	ORDER OF ROOT
(-2 +(-3)X + X**2)	48
(-2 +(-2)X + X**2)	48
(-2 +(2)X + X**2)	48
(-2 +(3)X + X**2)	48
(-1 +(-3)X + X**2)	16
(-1 +(-1)X + X**2)	16
(-1 +(1)X + X**2)	16
(-1 +(3)X + X**2)	16
(3 +(-2)X + X**2)	48
(3 +(-1)X + X**2)	48
(3 +(1)X + X**2)	48
(3 +(2)X + X**2)	48

TABLE OF ALL NON-ZERO ELEMENTS OVER GF(7**2) WITH PRIMITIVE POLYNOMIAL
P(X)=X**2 +(-1)*X + (3)

A** 0 =		1
A** 1 =	1A	
A** 2 =	1A +	-3
A** 3 =	-2A +	-3
A** 4 =	2A +	6
A** 5 =	1A +	-6
A** 6 =	2A +	-3
A** 7 =	6A +	-6
A** 8 =	0A +	-4
A** 9 =	3A +	0
A** 10 =	3A +	-2
A** 11 =	1A +	-2
A** 12 =	6A +	-3
A** 13 =	3A +	-4
A** 14 =	6A +	-2
A** 15 =	4A +	-4
A** 16 =	0A +	-5
A** 17 =	2A +	0
A** 18 =	2A +	-6
A** 19 =	3A +	-6
A** 20 =	4A +	-2
A** 21 =	2A +	-5

```
A** 22 =  4A +  -6
A** 23 =  5A +  -5
A** 24 =  0A +  -1
A** 25 =  6A +   0
A** 26 =  6A +  -4
A** 27 =  2A +  -4
A** 28 =  5A +  -6
A** 29 =  6A +  -1
A** 30 =  5A +  -4
A** 31 =  1A +  -1
A** 32 =  0A +  -3
A** 33 =  4A +   0
A** 34 =  4A +  -5
A** 35 =  6A +  -5
A** 36 =  1A +  -4
A** 37 =  4A +  -3
A** 38 =  1A +  -5
A** 39 =  3A +  -3
A** 40 =  0A +  -2
A** 41 =  5A +   0
A** 42 =  5A +  -1
A** 43 =  4A +  -1
```

```
A** 44 =   3A +  -5

A** 45 =   5A +  -2

A** 46 =   3A +  -1

A** 47 =   2A +  -2

A** 48 =   0A +  -6
```

```
GENERATOR POLYNOMIAL IS :

G(X) = X**6 +( -3)X**5 +( -1)X**4 +(  1)X**3 +( -3)X**2 +(  1)X +( -1)
```

NEGACYCLIC CODES, LEE METRIC, P = 7, R = 3, N = 171

X**171 + 1 =	ORDER OF ROOT
(-3 +(-3)X +(-3)X**2 + X**3)	342
(-3 +(3)X +(-3)X**2 + X**3)	342
(-3 +(-1)X +(-2)X**2 + X**3)	342
(-3 +(0)X +(-2)X**2 + X**3)	342
(-3 +(2)X +(-2)X**2 + X**3)	342
(-3 +(3)X +(-2)X**2 + X**3)	342
(-3 +(-3)X +(-1)X**2 + X**3)	342
(-3 +(-2)X +(-1)X**2 + X**3)	342
(-3 +(-1)X +(-1)X**2 + X**3)	342
(-3 +(0)X +(-1)X**2 + X**3)	342
(-3 +(0)X +(0)X**2 + X**3)	18
(-3 +(-2)X +(1)X**2 + X**3)	342
(-3 +(2)X +(1)X**2 + X**3)	342
(-3 +(-1)X +(2)X**2 + X**3)	342
(-3 +(1)X +(2)X**2 + X**3)	342
(-3 +(-2)X +(3)X**2 + X**3)	342
(-3 +(0)X +(3)X**2 + X**3)	342
(-3 +(1)X +(3)X**2 + X**3)	342
(-3 +(3)X +(3)X**2 + X**3)	342
(1 +(-1)X +(-3)X**2 + X**3)	114
(1 +(0)X +(-3)X**2 + X**3)	114
(1 +(3)X +(-3)X**2 + X**3)	38
(1 +(1)X +(-2)X**2 + X**3)	114
(1 +(2)X +(-2)X**2 + X**3)	114
(1 +(-3)X +(-1)X**2 + X**3)	114

(1 +(2)X +(-1)X**2 + X**3)	114
(1 +(-3)X +(0)X**2 + X**3)	114
(1 +(1)X +(0)X**2 + X**3)	114
(1 +(2)X +(0)X**2 + X**3)	38
(1 +(-2)X +(1)X**2 + X**3)	114
(1 +(0)X +(1)X**2 + X**3)	114
(1 +(3)X +(1)X**2 + X**3)	38
(1 +(-2)X +(2)X**2 + X**3)	114
(1 +(-1)X +(2)X**2 + X**3)	114
(1 +(0)X +(2)X**2 + X**3)	38
(1 +(-3)X +(3)X**2 + X**3)	38
(1 +(1)X +(3)X**2 + X**3)	38
(2 +(2)X +(-3)X**2 + X**3)	342
(2 +(3)X +(-3)X**2 + X**3)	342
(2 +(-3)X +(-2)X**2 + X**3)	342
(2 +(-2)X +(-2)X**2 + X**3)	342
(2 +(3)X +(-2)X**2 + X**3)	342
(2 +(-1)X +(-1)X**2 + X**3)	342
(2 +(1)X +(-1)X**2 + X**3)	342
(2 +(3)X +(-1)X**2 + X**3)	342
(2 +(-2)X +(0)X**2 + X**3)	342
(2 +(-1)X +(0)X**2 + X**3)	342
(2 +(0)X +(0)X**2 + X**3)	18
(2 +(3)X +(0)X**2 + X**3)	342
(2 +(-2)X +(1)X**2 + X**3)	342
(2 +(1)X +(1)X**2 + X**3)	342

(2 +(-3)X +(2)X**2 + X**3)	342
(2 +(-1)X +(2)X**2 + X**3)	342
(2 +(-2)X +(3)X**2 + X**3)	342
(2 +(-1)X +(3)X**2 + X**3)	342
(2 +(2)X +(3)X**2 + X**3)	342
(1 +(0)X +(0)X**2 +(-6)X**3	

TABLE OF ALL NON-ZERO ELEMENTS OVER GF(7**3) WITH PRIMITIVE POLYNOMIAL P(X)=X**3 +(3)*X**2 +(2)*X + (2)

A** 0 =						1
A** 1 =				1A		
A** 2 =	1A**2					
A** 3 =	-3A**2	+	-2A	+	-2
A** 4 =	0A**2	+	4A	+	6
A** 5 =	4A**2	+	6A	+	0
A** 6 =	-6A**2	+	-1A	+	-1
A** 7 =	3A**2	+	4A	+	5
A** 8 =	-5A**2	+	-1A	+	-6
A** 9 =	0A**2	+	4A	+	3
A** 10 =	4A**2	+	3A	+	0
A** 11 =	-2A**2	+	-1A	+	-1
A** 12 =	5A**2	+	3A	+	4
A** 13 =	-5A**2	+	-6A	+	-3
A** 14 =	2A**2	+	0A	+	3
A** 15 =	-6A**2	+	-1A	+	-4
A** 16 =	3A**2	+	1A	+	5
A** 17 =	-1A**2	+	-1A	+	-6
A** 18 =	2A**2	+	3A	+	2
A** 19 =	-3A**2	+	-2A	+	-4
A** 20 =	0A**2	+	2A	+	6
A** 21 =	2A**2	+	6A	+	0
A** 22 =	0A**2	+	-4A	+	-4
A** 23 =	-4A**2	+	3A	+	0
A** 24 =	1A**2	+	1A	+	1
A** 25 =	-2A**2	+	-1A	+	-2
A** 26 =	5A**2	+	2A	+	4

```
A**  27 =  -6A**2 +  -6A +  -3
A**  28 =   5A**2 +   2A +   5
A**  29 =  -6A**2 +  -5A +  -3
A**  30 =   6A**2 +   2A +   5
A**  31 =  -2A**2 +   0A +  -5
A**  32 =   6A**2 +   6A +   4
A**  33 =  -5A**2 +  -1A +  -5
A**  34 =   0A**2 +   5A +   3
A**  35 =   5A**2 +   3A +   0
A**  36 =  -5A**2 +  -3A +  -3
A**  37 =   5A**2 +   0A +   3
A**  38 =  -1A**2 +   0A +  -3
A**  39 =   3A**2 +   6A +   2
A**  40 =  -3A**2 +  -4A +  -6
A**  41 =   5A**2 +   0A +   6
A**  42 =  -1A**2 +  -4A +  -3
A**  43 =  -1A**2 +   6A +   2
A**  44 =   2A**2 +   4A +   2
A**  45 =  -2A**2 +  -2A +  -4
A**  46 =   4A**2 +   0A +   4
A**  47 =  -5A**2 +  -4A +  -1
A**  48 =   4A**2 +   2A +   3
A**  49 =  -3A**2 +  -5A +  -1
A**  50 =   4A**2 +   5A +   6
A**  51 =   0A**2 +  -2A +  -1
```

```
A**  52 =  -2A**2 +   6A +  0
A**  53 =   5A**2 +   4A +  4
A**  54 =  -4A**2 +  -6A + -3
A**  55 =   6A**2 +   5A +  1
A**  56 =  -6A**2 +  -4A + -5
A**  57 =   0A**2 +   0A +  5
A**  58 =   0A**2 +   5A +  0
A**  59 =   5A**2 +   0A +  0
A**  60 =  -1A**2 +  -3A + -3
A**  61 =   0A**2 +   6A +  2
A**  62 =   6A**2 +   2A +  0
A**  63 =  -2A**2 +  -5A + -5
A**  64 =   1A**2 +   6A +  4
A**  65 =   3A**2 +   2A + -2
A**  66 =   0A**2 +  -1A + -6
A**  67 =  -1A**2 +   1A +  0
A**  68 =   4A**2 +   2A +  2
A**  69 =  -3A**2 +  -6A + -1
A**  70 =   3A**2 +   5A +  6
A**  71 =  -4A**2 +   0A + -6
A**  72 =   5A**2 +   2A +  1
A**  73 =  -6A**2 +  -2A + -3
A**  74 =   2A**2 +   2A +  5
A**  75 =  -4A**2 +   1A + -4
A**  76 =   6A**2 +   4A +  1
A**  77 =   0A**2 +  -4A + -5
```

```
A**  78 = -4A**2 +  2A +  0
A**  79 =  0A**2 +  1A +  1
A**  80 =  1A**2 +  1A +  0
A**  81 = -2A**2 + -2A + -2
A**  82 =  4A**2 +  2A +  4
A**  83 = -3A**2 + -4A + -1
A**  84 =  5A**2 +  5A +  6
A**  85 = -3A**2 + -4A + -3
A**  86 =  5A**2 +  3A +  6
A**  87 = -5A**2 + -4A + -3
A**  88 =  4A**2 +  0A +  3
A**  89 = -5A**2 + -5A + -1
A**  90 =  3A**2 +  2A +  3
A**  91 =  0A**2 + -3A + -6
A**  92 = -3A**2 +  1A +  0
A**  93 =  3A**2 +  6A +  6
A**  94 = -3A**2 +  0A + -6
A**  95 =  2A**2 +  0A +  6
A**  96 = -6A**2 +  2A + -4
A**  97 =  6A**2 +  1A +  5
A**  98 = -3A**2 +  0A + -5
A**  99 =  2A**2 +  1A +  6
A** 100 = -5A**2 +  2A + -4
A** 101 =  3A**2 +  6A +  3
A** 102 = -3A**2 + -3A + -6
```

```
A** 103 =   6A**2 +   0A +  6
A** 104 =  -4A**2 +  -6A + -5
A** 105 =   6A**2 +   3A +  1
A** 106 =  -1A**2 +  -4A + -5
A** 107 =  -1A**2 +   4A +  2
A** 108 =   0A**2 +   4A +  2
A** 109 =   4A**2 +   2A +  0
A** 110 =  -3A**2 +  -1A + -1
A** 111 =   1A**2 +   5A +  6
A** 112 =   2A**2 +   4A + -2
A** 113 =  -2A**2 +   1A + -4
A** 114 =   0A**2 +   0A +  4
A** 115 =   CA**2 +   4A +  0
A** 116 =   4A**2 +   0A +  0
A** 117 =  -5A**2 +  -1A + -1
A** 118 =   0A**2 +   2A +  3
A** 119 =   2A**2 +   3A +  0
A** 120 =  -3A**2 +  -4A + -4
A** 121 =   5A**2 +   2A +  6
A** 122 =  -6A**2 +  -4A + -3
A** 123 =   0A**2 +   2A +  5
A** 124 =   2A**2 +   5A +  0
A** 125 =  -1A**2 +  -4A + -4
A** 126 =  -1A**2 +   5A +  2
A** 127 =   1A**2 +   4A +  2
A** 128 =   1A**2 +   0A + -2
```

A** 129 = -3A**2 + 3A + -2

A** 130 = 5A**2 + 4A + 6

A** 131 = -4A**2 + -4A + -3

A** 132 = 1A**2 + 5A + 1

A** 133 = 2A**2 + -1A + -2

A** 134 = 0A**2 + 1A + -4

A** 135 = 1A**2 + 3A + 0

A** 136 = 0A**2 + -2A + -2

A** 137 = -2A**2 + 5A + 0

A** 138 = 4A**2 + 4A + 4

A** 139 = -1A**2 + -4A + -1

A** 140 = -1A**2 + 1A + 2

A** 141 = 4A**2 + 4A + 2

A** 142 = -1A**2 + -6A + -1

A** 143 = -3A**2 + 1A + 2

A** 144 = 3A**2 + 1A + 6

A** 145 = -1A**2 + 0A + -6

A** 146 = 3A**2 + 3A + 2

A** 147 = -6A**2 + -4A + -6

A** 148 = 0A**2 + 6A + 5

A** 149 = 6A**2 + 5A + 0

A** 150 = -6A**2 + -5A + -5

A** 151 = 6A**2 + 0A + 5

A** 152 = -4A**2 + 0A + -5

A** 153 = 5A**2 + 3A + 1

A** 154 = -5A**2 + -2A + -3

A** 155 = 6A**2 + 0A + 3

A** 156 = -4A**2 + -2A + -5

A** 157 = 3A**2 + 3A + 1

A** 158 = -6A**2 + -5A + -6

A** 159 = 6A**2 + 6A + 5

A** 160 = -5A**2 + 0A + -5

A** 161 = 1A**2 + 5A + 3

A** 162 = 2A**2 + 1A + -2

A** 163 = -5A**2 + 1A + -4

A** 164 = 2A**2 + 6A + 3

A** 165 = 0A**2 + -1A + -4

A** 166 = -1A**2 + 3A + 0

A** 167 = 6A**2 + 2A + 2

A** 168 = -2A**2 + -3A + -5

A** 169 = 3A**2 + 6A + 4

A** 170 = -3A**2 + -2A + -6

A** 171 = 0A**2 + 0A + 6

A** 172 = 0A**2 + 6A + 0

A** 173 = 6A**2 + 0A + 0

A** 174 = -4A**2 + -5A + -5

A** 175 = 0A**2 + 3A + 1

A** 176 = 3A**2 + 1A + 0

A** 177 = -1A**2 + -6A + -6

A** 178 = -3A**2 + 3A + 2

A** 179 = 5A**2 + 1A + 6

A** 180 = 0A**2 + -4A + -3

A** 181 = -4A**2 + 4A + 0

A** 182 = 2A**2 + 1A + 1

A** 183 = -5A**2 + -3A + -4

A** 184 = 5A**2 + 6A + 3

A** 185 = -2A**2 + 0A + -3

A** 186 = 6A**2 + 1A + 4

A** 187 = -3A**2 + -1A + -5

A** 188 = 1A**2 + 1A + 6

A** 189 = -2A**2 + 4A + -2

A** 190 = 3A**2 + 2A + 4

A** 191 = 0A**2 + -2A + -6

A** 192 = -2A**2 + 1A + 0

A** 193 = 0A**2 + 4A + 4

A** 194 = 4A**2 + 4A + 0

A** 195 = -1A**2 + -1A + -1

A** 196 = 2A**2 + 1A + 2

A** 197 = -5A**2 + -2A + -4

A** 198 = 6A**2 + 6A + 3

A** 199 = -5A**2 + -2A + -5

A** 200 = 6A**2 + 5A + 3

A** 201 = -6A**2 + -2A + -5

A** 202 = 2A**2 + 0A + 5

A** 203 = -6A**2 + 1A + -4

A** 204 = 5A**2 + 1A + 5

```
A** 205 =   0A**2 +  -5A +  -3
A** 206 =  -5A**2 +   4A +   0
A** 207 =   5A**2 +   3A +   3
A** 208 =  -5A**2 +   0A +  -3
A** 209 =   1A**2 +   CA +   3
A** 210 =  -3A**2 +   1A +  -2
A** 211 =   3A**2 +   4A +   6
A** 212 =  -5A**2 +   0A +  -6
A** 213 =   1A**2 +   4A +   3
A** 214 =   1A**2 +   1A +  -2
A** 215 =  -2A**2 +   3A +  -2
A** 216 =   2A**2 +   2A +   4
A** 217 =  -4A**2 +   0A +  -4
A** 218 =   5A**2 +   4A +   1
A** 219 =  -4A**2 +  -2A +  -3
A** 220 =   3A**2 +   5A +   1
A** 221 =  -4A**2 +  -5A +  -6
A** 222 =   0A**2 +   2A +   1
A** 223 =   2A**2 +   1A +   0
A** 224 =  -5A**2 +  -4A +  -4
A** 225 =   4A**2 +   6A +   3
A** 226 =  -6A**2 +  -5A +  -1
A** 227 =   6A**2 +   4A +   5
A** 228 =   0A**2 +   0A +  -5
A** 229 =   0A**2 +   2A +   0
```

```
A** 230 =  2A**2 +  0A +  0
A** 231 = -6A**2 + -4A + -4
A** 232 =  0A**2 +  1A +  5
A** 233 =  1A**2 +  5A +  0
A** 234 =  2A**2 + -2A + -2
A** 235 = -1A**2 +  1A + -4
A** 236 =  4A**2 +  5A +  2
A** 237 =  0A**2 + -6A + -1
A** 238 = -6A**2 +  6A +  0
A** 239 =  3A**2 +  5A +  5
A** 240 = -4A**2 + -1A + -6
A** 241 =  4A**2 +  2A +  1
A** 242 = -3A**2 +  0A + -1
A** 243 =  2A**2 +  5A +  6
A** 244 = -1A**2 +  2A + -4
A** 245 =  5A**2 +  5A +  2
A** 246 = -3A**2 + -1A + -3
A** 247 =  1A**2 +  3A +  6
A** 248 =  0A**2 +  4A + -2
A** 249 =  4A**2 +  5A +  0
A** 250 =  0A**2 + -1A + -1
A** 251 = -1A**2 +  6A +  0
A** 252 =  2A**2 +  2A +  2
A** 253 = -4A**2 + -2A + -4
A** 254 =  3A**2 +  4A +  1
A** 255 = -5A**2 + -5A + -6
```

A** 256 =	3A**2	+	4A	+	3
A** 257 =	-5A**2	+	-3A	+	-6
A** 258 =	5A**2	+	4A	+	3
A** 259 =	-4A**2	+	0A	+	-3
A** 260 =	5A**2	+	5A	+	1
A** 261 =	-3A**2	+	-2A	+	-3
A** 262 =	0A**2	+	3A	+	6
A** 263 =	3A**2	+	6A	+	0
A** 264 =	-3A**2	+	-6A	+	-6
A** 265 =	3A**2	+	0A	+	6
A** 266 =	-2A**2	+	0A	+	-6
A** 267 =	6A**2	+	5A	+	4
A** 268 =	-6A**2	+	-1A	+	-5
A** 269 =	3A**2	+	0A	+	5
A** 270 =	-2A**2	+	-1A	+	-6
A** 271 =	5A**2	+	5A	+	4
A** 272 =	-3A**2	+	-6A	+	-3
A** 273 =	3A**2	+	3A	+	6
A** 274 =	-6A**2	+	0A	+	-6
A** 275 =	4A**2	+	6A	+	5
A** 276 =	-6A**2	+	-3A	+	-1
A** 277 =	1A**2	+	4A	+	5
A** 278 =	1A**2	+	3A	+	-2
A** 279 =	0A**2	+	3A	+	-2
A** 280 =	3A**2	+	5A	+	0
A** 281 =	-4A**2	+	-6A	+	-6

A** 282 = 6A**2 + 2A + 1

A** 283 = -2A**2 + -4A + -5

A** 284 = 2A**2 + 6A + 4

A** 285 = 0A**2 + 0A + -4

A** 286 = 0A**2 + 3A + 0

A** 287 = 3A**2 + 0A + 0

A** 288 = -2A**2 + -6A + -6

A** 289 = 0A**2 + 5A + 4

A** 290 = 5A**2 + 4A + 0

A** 291 = -4A**2 + -3A + -3

A** 292 = 2A**2 + 5A + 1

A** 293 = -1A**2 + -3A + -4

A** 294 = 0A**2 + 5A + 2

A** 295 = 5A**2 + 2A + 0

A** 296 = -6A**2 + -3A + -3

A** 297 = 1A**2 + 2A + 5

A** 298 = -1A**2 + 3A + -2

A** 299 = 6A**2 + 0A + 2

A** 300 = -4A**2 + -3A + -5

A** 301 = 2A**2 + 3A + 1

A** 302 = -3A**2 + -3A + -4

A** 303 = 6A**2 + 2A + 6

A** 304 = -2A**2 + -6A + -5

A** 305 = 0A**2 + 6A + 4

A** 306 = 6A**2 + 4A + 0

```
A** 307 =   0A**2 +  -5A +  -5
A** 308 =  -5A**2 +   2A +   0
A** 309 =   3A**2 +   3A +   3
A** 310 =  -6A**2 +  -3A +  -6
A** 311 =   1A**2 +   6A +   5
A** 312 =   3A**2 +   3A +  -2
A** 313 =  -6A**2 +  -1A +  -6
A** 314 =   3A**2 +   6A +   5
A** 315 =  -3A**2 +  -1A +  -6
A** 316 =   1A**2 +   0A +   6
A** 317 =  -3A**2 +   4A +  -2
A** 318 =   6A**2 +   4A +   6
A** 319 =   0A**2 +  -6A +  -5
A** 320 =  -6A**2 +   2A +   0
A** 321 =   6A**2 +   5A +   5
A** 322 =  -6A**2 +   0A +  -5
A** 323 =   4A**2 +   0A +   5
A** 324 =  -5A**2 +  -3A +  -1
A** 325 =   5A**2 +   2A +   3
A** 326 =  -6A**2 +   0A +  -3
A** 327 =   4A**2 +   2A +   5
A** 328 =  -3A**2 +  -3A +  -1
A** 329 =   6A**2 +   5A +   6
A** 330 =  -6A**2 +  -6A +  -5
A** 331 =   5A**2 +   0A +   5
A** 332 =  -1A**2 +  -5A +  -3
```

```
A** 333 =   -2A**2 +   6A +   2
A** 334 =    5A**2 +   6A +   4
A** 335 =   -2A**2 +  -6A +  -3
A** 336 =    0A**2 +   1A +   4
A** 337 =    1A**2 +   4A +   0
A** 338 =    1A**2 +  -2A +  -2
A** 339 =   -5A**2 +   3A +  -2
A** 340 =    4A**2 +   1A +   3
A** 341 =   -4A**2 +  -5A +  -1
A** 342 =    0A**2 +   0A +   1
```

GENERATOR POLYNOMIAL IS :

```
G(X) = X**9 +(  2)X**8 +(  2)X**7 +( -3)X**6 +( -2)X**5
     +(  1)X**4 +(  2)X**3 +(  0)X**2 +( -1)X +(  1)
```

NEGACYCLIC CODES, LEE METRIC, P = 11,R = 2, N = 60

X** 60 + 1 =	ORDER OF ROOT
(-5 +(-3)X + X**2)	120
(-5 +(-2)X + X**2)	120
(-5 +(-1)X + X**2)	40
(-5 +(1)X + X**2)	40
(-5 +(2)X + X**2)	120
(-5 +(3)X + X**2)	120
(-4 +(-5)X + X**2)	40
(-4 +(-4)X + X**2)	120
(-4 +(-1)X + X**2)	120
(-4 +(1)X + X**2)	120
(-4 +(4)X + X**2)	120
(-4 +(5)X + X**2)	40
(-3 +(-4)X + X**2)	40
(-3 +(-3)X + X**2)	120
(-3 +(-1)X + X**2)	120
(-3 +(1)X + X**2)	120
(-3 +(3)X + X**2)	120
(-3 +(4)X + X**2)	40
(-1 +(-5)X + X**2)	24
(-1 +(-3)X + X**2)	8
(-1 +(-2)X + X**2)	24
(-1 +(2)X + X**2)	24
(-1 +(3)X + X**2)	8
(-1 +(5)X + X**2)	24
(2 +(-5)X + X**2)	120

```
(  2 +( -4)X + X**2 )          120
(  2 +( -2)X + X**2 )           40
(  2 +(  2)X + X**2 )           40
(  2 +(  4)X + X**2 )          120
(  2 +(  5)X + X**2 )          120
```

TABLE OF ALL NON-ZERO ELEMENTS OVER GF(7**2) WITH PRIMITIVE POLYNOMIAL

P(X)=X**2 +(-4)*X + (2)

A**	0 =			1
A**	1 =	1A		
A**	2 =	4A	+	-2
A**	3 =	3A	+	-8
A**	4 =	4A	+	-6
A**	5 =	10A	+	-8
A**	6 =	10A	+	-9
A**	7 =	9A	+	-9
A**	8 =	5A	+	-7
A**	9 =	2A	+	-10
A**	10 =	9A	+	-4
A**	11 =	10A	+	-7
A**	12 =	0A	+	-9
A**	13 =	2A	+	0
A**	14 =	8A	+	-4
A**	15 =	6A	+	-5
A**	16 =	8A	+	-1
A**	17 =	9A	+	-5
A**	18 =	9A	+	-7
A**	19 =	7A	+	-7
A**	20 =	10A	+	-3
A**	21 =	4A	+	-9
A**	22 =	7A	+	-8
A**	23 =	9A	+	-3
A**	24 =	0A	+	-7
A**	25 =	4A	+	0

```
A** 26 =  5A +  -8
A** 27 =  1A + -10
A** 28 =  5A +  -2
A** 29 =  7A + -10
A** 30 =  7A +  -3
A** 31 =  3A +  -3
A** 32 =  9A +  -6
A** 33 =  8A +  -7
A** 34 =  3A +  -5
A** 35 =  7A +  -6
A** 36 =  0A +  -3
A** 37 =  8A +   0
A** 38 = 10A +  -5
A** 39 =  2A +  -9
A** 40 = 10A +  -4
A** 41 =  3A +  -9
A** 42 =  3A +  -6
A** 43 =  6A +  -6
A** 44 =  7A +  -1
A** 45 =  5A +  -3
A** 46 =  6A + -10
A** 47 =  3A +  -1
A** 48 =  0A +  -6
A** 49 =  5A +   0
A** 50 =  9A + -10
A** 51 =  4A +  -7
```

A** 52 = 9A + -8

A** 53 = 6A + -7

A** 54 = 6A + -1

A** 55 = 1A + -1

A** 56 = 3A + -2

A** 57 = 10A + -6

A** 58 = 1A + -9

A** 59 = 6A + -2

A** 60 = 0A + -1

A** 61 = 10A + 0

A** 62 = 7A + -9

A** 63 = 8A + -3

A** 64 = 7A + -5

A** 65 = 1A + -3

A** 66 = 1A + -2

A** 67 = 2A + -2

A** 68 = 6A + -4

A** 69 = 9A + -1

A** 70 = 2A + -7

A** 71 = 1A + -4

A** 72 = 0A + -2

A** 73 = 9A + 0

A** 74 = 3A + -7

A** 75 = 5A + -6

A** 76 = 3A + -10

A** 77 = 2A + -6

```
A** 78 =  2A + -4
A** 79 =  4A + -4
A** 80 =  1A + -8
A** 81 =  7A + -2
A** 82 =  4A + -3
A** 83 =  2A + -8
A** 84 =  0A + -4
A** 85 =  7A +  0
A** 86 =  6A + -3
A** 87 = 10A + -1
A** 88 =  6A + -9
A** 89 =  4A + -1
A** 90 =  4A + -8
A** 91 =  8A + -8
A** 92 =  2A + -5
A** 93 =  3A + -4
A** 94 =  8A + -6
A** 95 =  4A + -5
A** 96 =  0A + -8
A** 97 =  3A +  0
A** 98 =  1A + -6
A** 99 =  9A + -2
A**100 =  1A + -7
A**101 =  8A + -2
A**102 =  8A + -5
```

```
A**103 =   5A +  -5

A**104 =   4A + -10

A**105 =   6A +  -8

A**106 =   5A +  -1

A**107 =   8A + -10

A**108 =   0A +  -5

A**109 =   6A +   0

A**110 =   2A +  -1

A**111 =   7A +  -4

A**112 =   2A +  -3

A**113 =   5A +  -4

A**114 =   5A + -10

A**115 =  10A + -10

A**116 =   8A +  -9

A**117 =   1A +  -5

A**118 =  10A +  -2

A**119 =   5A +  -9

A**120 =   0A + -10
```

GENERATOR POLYNOMIAL IS :

```
G(X) = X**10 +(  3)X**9 +(  0)X**8 +(  5)X**7 +(  3)X**6
     +(  2)X**5 +( -5)X**4 +( -2)X**3 +(  0)X**2 +(  4)X +( -1)
```

NEGACYCLIC CODES, LEE METRIC, P = 13, R = 2, N = 84

X** 84 + 1 =	ORDER OF ROOT
(-6 +(-6)X + X**2)	168
(-6 +(-3)X + X**2)	168
(-6 +(-2)X + X**2)	168
(-6 +(0)X + X**2)	24
(-6 +(2)X + X**2)	168
(-6 +(3)X + X**2)	168
(-6 +(6)X + X**2)	168
(-5 +(-5)X + X**2)	56
(-5 +(-2)X + X**2)	56
(-5 +(-1)X + X**2)	56
(-5 +(0)X + X**2)	8
(-5 +(1)X + X**2)	56
(-5 +(2)X + X**2)	56
(-5 +(5)X + X**2)	56
(-2 +(-6)X + X**2)	168
(-2 +(-5)X + X**2)	168
(-2 +(-4)X + X**2)	168
(-2 +(0)X + X**2)	24
(-2 +(4)X + X**2)	168
(-2 +(5)X + X**2)	168
(-2 +(6)X + X**2)	168
(2 +(-6)X + X**2)	168
(2 +(-4)X + X**2)	168
(2 +(-1)X + X**2)	168
(2 +(0)X + X**2)	24

(2 +(1)X + X**2)	168
(2 +(4)X + X**2)	168
(2 +(6)X + X**2)	168
(5 +(-5)X + X**2)	56
(5 +(-3)X + X**2)	56
(5 +(-1)X + X**2)	56
(5 +(0)X + X**2)	8
(5 +(1)X + X**2)	56
(5 +(3)X + X**2)	56
(5 +(5)X + X**2)	56
(6 +(-4)X + X**2)	168
(6 +(-3)X + X**2)	168
(6 +(-2)X + X**2)	168
(6 +(0)X + X**2)	24
(6 +(2)X + X**2)	168
(6 +(3)X + X**2)	168
(6 +(4)X + X**2)	168

TABLE OF ALL NON-ZERO ELEMENTS OVER GF(7**2) WITH PRIMITIVE POLYNOMIAL

P(X)=X**2 +(-6)*X + (-6)

A** 0 =				1
A** 1 =	1A			
A** 2 =	6A	+		6
A** 3 =	3A	+		10
A** 4 =	2A	+		5
A** 5 =	4A	+		12
A** 6 =	10A	+		11
A** 7 =	6A	+		8
A** 8 =	5A	+		10
A** 9 =	1A	+		4
A** 10 =	10A	+		6
A** 11 =	1A	+		8
A** 12 =	1A	+		6
A** 13 =	12A	+		6
A** 14 =	0A	+		7
A** 15 =	7A	+		0
A** 16 =	3A	+		3
A** 17 =	8A	+		5
A** 18 =	1A	+		9
A** 19 =	2A	+		6
A** 20 =	5A	+		12
A** 21 =	3A	+		4
A** 22 =	9A	+		5
A** 23 =	7A	+		2
A** 24 =	5A	+		3
A** 25 =	7A	+		4
A** 26 =	7A	+		3

A** 27 = 6A + 3

A** 28 = 0A + 10

A** 29 = 10A + 0

A** 30 = 8A + 8

A** 31 = 4A + 9

A** 32 = 7A + 11

A** 33 = 1A + 3

A** 34 = 9A + 6

A** 35 = 8A + 2

A** 36 = 11A + 9

A** 37 = 10A + 1

A** 38 = 9A + 8

A** 39 = 10A + 2

A** 40 = 10A + 8

A** 41 = 3A + 8

A** 42 = 0A + 5

A** 43 = 5A + 0

A** 44 = 4A + 4

A** 45 = 2A + 11

A** 46 = 10A + 12

A** 47 = 7A + 8

A** 48 = 11A + 3

A** 49 = 4A + 1

A** 50 = 12A + 11

A** 51 = 5A + 7

```
A** 52 = 11A + 4
A** 53 =  5A + 1
A** 54 =  5A + 4
A** 55 =  8A + 4
A** 56 =  ØA + 9
A** 57 =  9A + Ø
A** 58 =  2A + 2
A** 59 =  1A + 12
A** 60 =  5A + 6
A** 61 = 1ØA + 4
A** 62 = 12A + 8
A** 63 =  2A + 7
A** 64 =  6A + 12
A** 65 =  9A + 1Ø
A** 66 = 12A + 2
A** 67 =  9A + 7
A** 68 =  9A + 2
A** 69 =  4A + 2
A** 70 =  ØA + 11
A** 71 = 11A + Ø
A** 72 =  1A + 1
A** 73 =  7A + 6
A** 74 =  9A + 3
A** 75 =  5A + 2
A** 76 =  6A + 4
A** 77 =  1A + 1Ø
```

A** 78 = 3A + 6
A** 79 = 11A + 5
A** 80 = 6A + 1
A** 81 = 11A + 10
A** 82 = 11A + 1
A** 83 = 2A + 1
A** 84 = 0A + 12
A** 85 = 12A + 0
A** 86 = 7A + 7
A** 87 = 10A + 3
A** 88 = 11A + 8
A** 89 = 9A + 1
A** 90 = 3A + 2
A** 91 = 7A + 5
A** 92 = 8A + 3
A** 93 = 12A + 9
A** 94 = 3A + 7
A** 95 = 12A + 5
A** 96 = 12A + 7
A** 97 = 1A + 7
A** 98 = 0A + 6
A** 99 = 6A + 0
A**100 = 10A + 10
A**101 = 5A + 8

A**102	=	12A	+	4
A**103	=	11A	+	7
A**104	=	8A	+	1
A**105	=	10A	+	9
A**106	=	4A	+	8
A**107	=	6A	+	11
A**108	=	8A	+	10
A**109	=	6A	+	9
A**110	=	6A	+	10
A**111	=	7A	+	10
A**112	=	0A	+	3
A**113	=	3A	+	0
A**114	=	5A	+	5
A**115	=	9A	+	4
A**116	=	6A	+	2
A**117	=	12A	+	10
A**118	=	4A	+	7
A**119	=	5A	+	11
A**120	=	2A	+	4
A**121	=	3A	+	12
A**122	=	4A	+	5
A**123	=	3A	+	11
A**124	=	3A	+	5
A**125	=	10A	+	5
A**126	=	0A	+	8
A**127	=	8A	+	0

A**128 = 9A + 9

A**129 = 11A + 2

A**130 = 3A + 1

A**131 = 6A + 5

A**132 = 2A + 10

A**133 = 9A + 12

A**134 = 1A + 2

A**135 = 8A + 6

A**136 = 2A + 9

A**137 = 8A + 12

A**138 = 8A + 9

A**139 = 5A + 9

A**140 = 0A + 4

A**141 = 4A + 0

A**142 = 11A + 11

A**143 = 12A + 1

A**144 = 8A + 7

A**145 = 3A + 9

A**146 = 1A + 5

A**147 = 11A + 6

A**148 = 7A + 1

A**149 = 4A + 3

A**150 = 1A + 11

A**151 = 4A + 6

A**152 = 4A + 11

A**153 = 9A + 11

```
A**154 =   0A +  2
A**155 =   2A +  0
A**156 =  12A + 12
A**157 =   6A +  7
A**158 =   4A + 10
A**159 =   8A + 11
A**160 =   7A +  9
A**161 =  12A +  3
A**162 =  10A +  7
A**163 =   2A +  8
A**164 =   7A + 12
A**165 =   2A +  3
A**166 =   2A + 12
A**167 =  11A + 12
A**168 =   0A +  1
```

GENERATOR POLYNOMIAL IS :

```
G(X) = X**12 + (  2)X**11 +(  6)X**10 +( -1)X**9 +( -3)X**8
       +(  4)X**7 +(  0)X**6 +( -2)X**5 +(  4)X**4 +(  5)X**3
       +( -2)X**2 +(  4)X +(  1)
```

References

[1] D. Estournet, "Compression d'information de signaux d'images par les systèmes différentiels codés", L'onde électrique, vol. 49, fasc. 8, septembre 1969, p. 858.

[2] S.G.S. Shiva, J.-P. Dion, V.C. Chau, "Review of Digital Image Processing Techniques for Bandwidth Reduction", preliminary report submitted to the Dept. of Communications under contract No 0LGR-36100-2-0208; Dec. 15, 1973. Chap. 3.

[3] M. Eleccion, "A/D and D/A Converters", IEEE Spectrum, Vol. 9, No. 7, July 1972, p. 63.

[4] Shiva, Dion, Chau, ibid, Chap. 2 and Chap. 3.

[5] P.J. Ready, P.A. Wintz, "Information Extraction, SNR Improvement, and Data Compression in Multispectral Imagery", IEEE Transactions on Communications, Vol. COM-21, No 10, October 1973, p. 1123.

[6] B. Fino, "Etude expérimentale du codage d'images par les transformations de Haar et de Hadamard complexe", Annales des Télécommunications, Tome 27, No. 5-6, mai-juin 1972, pp 185-208.

[7] J.B. O'Neal, Jr., "Predictive Quantizing Systems (Differential Pulse Code Modulation) for the Transmission of Television Signals", BSTJ, May-June 1966, pp. 689-695.

[8] R.E. Graham, "Predictive Quantizing of Television Signals", IRE Wescon Convention Record, Part 4, Aug. 1958, pp 147-156.

[9] R.J. Arguello, H.R. Sellner, J.A. Stuller, "The Effect of Channel Errors in the Differential Pulse Code Modulation Transmission of Sampled Imagery", IEEE Transactions on Communication Technology, Vol. COM -19, No. 6, Dec. 1971, pp 926-933.

[10] J.B. O'Neal, Jr., ibid, p. 692.

[11] A.V.J. Martin, "Two-Dimensional Predictive Redundancy in a Television Display", IRE Transactions on Communications Systems, May 1959, p. 58.

[12] J.W. Bayless, S.J. Campanella, A.J. Goldberg, "Voice signals: bit-by-bit", IEEE Spectrum, October 1973, pp. 28-34.

[13] R.F. Rice, J.R. Plaunt, "Adaptive Variable-Length Coding for Efficient Compression of Spacecraft Television Data", IEEE Transactions on Communication Technology, Vol. COM-19, No. 6, December 1971, p. 890.

[14] D. Estournet, ibid, p. 861.

[15] D. Estournet, "Etude statistique d'un signal d'image", L'Onde électrique, vol. 49, fasc. 8, septembre 1969, pp. 848-852.

[16] D. Estournet, "Compression d'information de signaux d'images par les systèmes différentiels codés", L'onde électrique, Vol. 49, fasc. 8, septembre 1969, pp. 858-867.

[17] J.C. Jolivet, F.-X. Stouls, "Codage différentiel d'images multiniveaux, application au visiophone", Annales des Télécommunications, t. 27, Nos. 5-6, 1972, pp 229-230.

[18] E.R. Berlekamp, "Algebraic Coding Theory", McGraw-Hill, New York, 1968, pp. 200-217.

[19] V.C. Chau, "Study of Negacyclic Codes", M.A.Sc. thesis, University of Ottawa, February, 1974.