Optimal Space Communication Techniques

Final Report

June 15, 1974 - August 14, 1975

Goddard Space Flight Center

Greenbelt, Maryland

under

NASA Grant: NSG - 5013

Donald L. Schilling

Professor of Electrical Engineering

Principal Investigator

COMMUNICATIONS SYSTEMS LABORATORY

DEPARTMENT OF ELECTRICAL ENGINEERING

THE CITY COLLEGE OF
THE CITY UNIVERSITY of NEW YORK

Table of Contents

Introduction

This final report summarizes several aspects of the research sponsored by the National Aeronautics and Space Administration under NASA Grant NSG - 5013 for the period June 15, 1974 - August 14, 1975. The research discussed in this report considers the "Encoding of Video Signals using Adaptive Delta Modulation", "Conversion of PCM to DM" and an Introduction to a new study area - "A Phase Locked Loop Using a Nonlinear Processor in lieu of a Loop Filter".

An adaptive delta modulator has been constructed using the Song Delta Modulator and many results have been presented in prior reports (see the June 1975 - December 1975 Semiannual Report). Part I of this final report concludes our study of error correction of DM encoded signals corrupted by thermal noise. Future research in this area will focus on real time encoding of black-and-white and of color video signals. We are also beginning to look at two-dimensional and frame-to-frame encoding. A real time DM has been constructed and the memory needed to permit frame-to-frame encoding is being assembled. In addition, we are looking at source-encoding techniques with which to compress the DM encoded bit stream. This approach appears to be promising in as much as the DM bit stream provides sequences which are not equally likely. A discussion of this project will be made in our next report. A new area in which research is beginning is interference of RF signals containing DM encoded information. In this study we are looking at the effect of an interfering CW, FM, or other interfering signal with a PSK, DPSK, QPSK or FSK signal where the information being sent is a DM encoded bit stream.

Part II of this report "Conversion from PCM to Adaptive DM" is a study of how to "fill in" extra samples of a PCM signal so that an adaptive DM bit stream can be obtained. In this report we derive an expression for the SNR of the DM signal derived by employing linear, 2-sample, interpolation between

sample points. We are currently investigating 3 - and higher sample interpolation techniques.

Part III is the introduction to a new study of PLL acquisition. Here we replace the linear loop filter by a nonlinear processor. Estimation Theory tells us that the optimum estimator in a PLL is a linear filter when the PLL is locked. However, when the loop is not locked the effective noise emanating from the phase detector is not gaussian. Thus, a nonlinear processor can provide more efficient acquisition. The purpose of a loop filter is to provide a voltage to the VCO so that it can correct its frequency and the nonlinear processor also provides this voltage. However, the algorithm employed by the nonlinear processor differs from the algorithm employed by the loop filter. As a matter of fact, we employ the SONG DM algorithm so that we tend to acquire exponentially. We intend to employ overshoot suppression algorithms to prevent overshoot once we have acquired.

During the grant period several papers were presented at conferences and one is to be published. A complete list of publications and presentations is given in Part IV.

Participating in this project are:

Drs. J. Garodnick and D. L. Schilling

and Doctoral Students:

R. Lei, J. LoCicero, V. Rao, N. Scheinberg, D. Ucci and L. Weiss.

Part I Encoding of Video Signals Using Adaptive Delta Modulation

Introduction

A delta modulator algorithm is described. The delta modulator algorithm is then used to encode pictures and the encoded pictures are compared to the same pictures encoded by PCM. The effects of channel errors on the delta modulated video signal are explained and several error correction algorithms are explored.

The Delta Modulator

The delta modulator used to encode the pictures in this paper is shown in Fig. 1. The equations describing the operations of the delta modulator are given below:

$$E_k = \text{Sgn}\,(S_k - X_k) \qquad (1)$$

$$X_{k+1} = X_k + \Delta_{k+1} \qquad (2)$$

$$\Delta_{k+1} = \begin{cases} |\Delta_k|\,(E_k + \frac{1}{2}E_{k-1}); & |\Delta_k| \geq 2\Delta \\ 2\Delta E_k & |\Delta_k| < 2\Delta \end{cases} \qquad (3)$$

$E_k \equiv$ Output of the encoder

$S_k \equiv$ Input to the encoder

$X_k \equiv$ The encoders estimate of the input signal at the K^{th} instant of time; also, the decoders output at the K^{th} instant of time

$\Delta_k \equiv$ The step size of the delta modulator

$\Delta \equiv$ A constant that determines the minimum allowable value for Δ_k

There are other delta modulator algorithms besides the one described by Eq. 1, 2 and 3. In general the differences lie in Eq. 3, the way the step size is formed. In some delta modulators $|\Delta_k|$ is a constant. In others it increases or decreases linearly. In our delta modulator the step size, Δ_k, changes exponentially. It has been experimentally determined

that an exponentially changing step size type of delta modulator is more accurate then the other types of delta modulators at encoding a signal with large step type changes in amplitude found in video signals. Experimental work done by us, Cuttler [1] and others, as well as theoretical studies by Song [2] has shown that for video signals the step size should grow by a factor of about 1.5 and decrease by a factor of .5. The delta modulator described by Eq. 3 behaves this way.

The delta modulator shown in Fig. 1 is not only a good encoder of video signals as compared with some other delta modulators, but it is also simple to impliment. From Fig. 1 we see that the decoder is the same as the feedback loop of the encoder. Delay elements D_1 and D_2 are each a single flip flop and the two multipliers are just sign gates (exclusive or gates). The multiplication by $\frac{1}{2}$ and the formation of the absolute value of Δ_k is a wired operation and requires no additional hardware to impliment.

The Step Response

The response of the delta modulator to a step like input is shown in Fig. 2. The edges of objects in pictures are like steps; hence, examination of Fig. 2 will help explain what happens to edges in pictures that have been incoded by a delta modulator. Note from Fig. 2 that at first the delta modulator's output does not rise as fast as the input signal. This effect is called slope overload. Slope overload on the edges of objects has the effect of delaying the appearance of the edge in the picture. After the delta modulator "catches up" with the input signal it often overshoots the input and then settles down to the repeditive four bit "steady state" pattern shown on the horizontal portion of the step. If the delta modulator is sampling the input signal at a rate greater than twice the nyquest rate, then all the frequency components in the overshoots and the steady state pattern will be higher

than the highest frequency components in the input signal; hence they may be removed by low pass filtering at the output of the receiver without degrading the picture.

The dotted line in Fig. 2 represents the response of the same delta modulator to the same step like input but the delta modulator had different initial conditions at the start of the step. These initial conditions resulted in a minimum of "slope overload" and would result in almost no delay in the appearance of the edge in the picture. The effect of delaying an edge in the picture sometimes, but not at other times, causes all sharp edges at right angles to the direction of scan to wiggle. This effect is called edge business. Edge business is the most serious hindrance to encoding high quality pictures at low bit rates with delta modulators.

PCM vs Delta Modulated Pictures

From Fig. 3 a comparision can be made between delta modulated pictures and 64 quantization level PCM encoded pictures. Note at high bit rates (682 bits/line) both PCM and delta modulation produce satisfactory pictures. At low bit rates (410 bits/line) edge business can be seen in the delta modulated pictures and a loss of resolution can be seen in the PCM encoded pictures. The edge business shows up as a wiggliness along the right edge of my face, and the loss of resolution in the PCM encoded pictures is apparent by the loss of the stripes on my shirt. A subjective evaluation of Fig. 3 leads us to the conclusion that at high bit rates the PCM encoding is desirable and at low bit rates delta modulation perserves more detail in the picture then PCM.

The Effects of Channel Errors on Delta Modulated Signals

The response of the delta modulator decoder to a step input at the encoder in the presence of a single channel error is shown in Fig. 4. The solid lines represent the output signal, X_k, of the decoder in the absence of channel errors, while the dashed lines represent the output signal of the decoder in the presents of a single channel error.

Channel errors have two effects on the delta modulator. The first and most obvious effect that can be seen in Fig. 4 is that channel errors always cause a permanent, and usually large DC shift in the received signal. Less obvious is the fact that a single channel error may cause an increase, decrease or have no effect at all on the step size of the delta modulator decoder [ ], and less obvious still, is the fact that step size errors, when they occur, become self correcting within a few samples. Because step size errors last for only a few samples, and delta modulators typically sample at several times the nyquest rate of the input signal, the disturbance caused by step size errors is usually one to two pixels long and can bearly be perceived in the picture.

We have both quantitative results and experimental evidence to support the statements made in the preceeding paragraph. The quantitative results were arrived at through the following process. Observe from Fig. 4 and Eqs. 1, 2 and 3 that the step size, Δ_k, will decrease while the delta modulator is tracking a constant DC level. If the constant signal level persists long enough the delta modulator reaches the minimum step size, Δ, whether or not a channel error has occured. When the delta modulator reaches the minimum step size, the step size error has corrected itself, since both the corrupted signal and the ideal error free signal would have the same step size namely Δ, the minimum step size.

The number of transmitted bits, (e_k's), that will reach the receiver after the occurrence of a channel error, but before the step size corrects itself, has an upper bound for the delta modulator tracking a constant DC level. This upper bound can be shown to be

$$.75^{n/2} = \Delta/\Delta_{k+1}$$

$$\text{or} \quad N = \frac{2 \ln \Delta / \Delta_{k+1}}{\ln .75} \qquad (4)$$

where

$N \equiv$ the number of transmitted bits until step size correction occurs

$\Delta_{k+1} \equiv$ the step size one sample time after the occurence of the error

$\Delta \equiv$ the minimum allowable value for Δ_k

For the pictures in this paper the worst case parameters for Eq. 4 are $\Delta = 1$ and $\Delta_{k+1} = 16$ which yield a worst case $N = 19$. If the delta modulator samples at six times the nyquest rate of the video signal then the step size error should last for less than 3 pixels if the conditions imposed in deriving Eq. 4 hold for a real picture.

The series of pictures in Fig. 5 were taken to confirm the results of Eq. 4, ie, step size errors quickly correct themselves, and to comfirm that channel errors cause a large permanent shift in the DC level of the output of the delta modulator. Fig. 5a shows the original picture without any errors. In Fig. 5b channel errors were introduced at a rate of one error for every 2,500 transmitted bits. From Fig. 5b we see that each channel error caused a shift in the level of the decoders output signal. This shift, seen as a streak, lasts until the end of the scanning line where the effect of the error is ended by resetting all the registers in the encoder and decoder to a fixed predetermined value.

In Fig. 5c we have displayed the absolute value of the step size without any channel errors and in Fig. 5d we introduced channel errors. Fig. 5e is the difference between Fig. 5c and Fig. 5d. The uniform gray background in Fig. 5e represents zero difference between Fig. 5c and Fig. 5d or equivently, no step size error. The white and black dots are the regions where the difference between Fig. 5c and Fig. 5d was not zero or equivently, where a step size error exists. Fig. 5e confirms the fact that step size errors become self correcting after a few samples. The conclusions that we have drawn from our studies of the effects of channel errors on delta modulated video signals is that channel errors have a significent effect only on the estimate, X_k, of the delta modulator decoder and that error correcting schemes need only correct for errors in the decoder's estimate, X_k.

Error Correcting Algorithms

The preceeding analysis has revealed that channel errors effect delta modulated encoded video signals by changing the DC level of the decoded signal. In the rest of this paper we will describe three techniques to minimize this effect.

Direct Approach

It is possible to correct the DC level of the decoders estimate by periodically sending the transmitters current estimate, X_k, to the receiver. The effect of this correction technique, (shown in Fig. 6) is to shorten the length of the error streaks. The more often the transmitters estimate is sent to the receiver the shorter the streaks become. Unfortunately the

more often we send the transmitters estimate, the more we must increase the transmission rate to accomodate this extra information.

The equation that relates the several parameters that determine the increase in transmission rate is given by Eq. 5.

$$f_s' = f\left(1 + \frac{cb}{s}\right) \qquad (5)$$

where

f_s' ≡ The bit rate with correction

f ≡ The bit rate without correction

c ≡ The number of times X_k is sent to the receiver per frame

b ≡ The number of bits in each X_k sent to the receiver

s ≡ The total number of E_k's transmitted per frame

From Eq. 5 it is apparent that "c" and "b" should be as small as possible. "c" is lower bounded by the desired degree of correction required in the picture since the length of the remaining error streaks in the received picture are inversely proportional to c, and b is lower bounded by the accuracy of the correction.

To understand the effect of b on the correction algorithm refer to Fig. 7. Fig. 7 shows how the correction algorithm is implimented. First the "b" most significent bits of the transmitters estimate is sent to the receiver via a PCM format. Upon receiving the PCM word the receiver sets the "b" most significent bits in its estimate equal to the transmitters estimate. Then both the transmitter and receiver set their b-1 most significent bit to 1 and all less significent bits to zero. The total effect is to make the transmitter's and receiver's estimate equal, and to entroduce an inacuracy in the estimates equal to 2^{-b-1} percent.

The effect of the inaccuracy in the estimates introduced by the correction algorithm is shown in Fig. 8. Note that the effect is transient and is completely undetectable for b = 4. The value b = 4 was used in the pictures of Fig. 6.

Leaky Integrator

The effects of channel errors on delta modulated encoded pictures can be minimized by introducing leaky integrators in the feed back loop of the encoder and in the decoder. Leaky integration is achieved by introducing the factor L, ($L \leq 1$), into Eq. 2 as shown in Eq. 6.

$$X_{K+1} = LX_K + \Delta_{K+1} \qquad (6)$$

If a channel error causes the estimate, X_K, to become shifted by an amount "e" at time "k", then at time K + N Eq. 6 will become

$$X_{K+N} = L^N X_K + \left\{ \sum_{i=1}^{N} L^{N-i} \Delta_{K+i} \right\} + L^N e \qquad (7)$$

From Eq. 7 we see that the error "e" will leak away by the factor L, and after N samples the amplitude of the error will be $L^N e$. Clearly, the smaller the value of L the sooner the error will disappear. The smallest value of L that may be used without degrading a picture with 64 quantization levels ranging from -32 to +31 is L = .967.

Fig. 9 shows the effect of using a leaky integrator with L = .967 on delta modulated encoded pictures that were transmitted over a noisy channel. From Fig. 9 we see that the inclusion of leaky integration in our delta modulator will substantially reduce the effects of channel errors on the received pictures.

Line to Line Correlation

In this section we will explain a technique, shown in Fig. 10, which locates error streaks in pictures and which then eliminates the streaks by readjusting the DC level of the streak to its proper value.

To detect an error streak in a picture a comparision is made between the DC level of the portion of a scan line under test and the corresponding portion of the scan line above and below the line under test. If the DC level of the portion of the line under test differs from both the line above and below by more than a certain "threshold" then an error has been detected. When an error is detected the DC level of the portion of the line under test is replaced by the average DC value of the corresponding portion of the line above and below. The part of the picture referred to as "the portion under test" is formed from "N" consective samples on a scaning line. If the first sample in the "N" samples is "i", then the algorithm indexes its "portion under test" by including the i+N+1 sample and dropping the i^{th} sample.

There are two parameters in the above algorithm that must be adjusted, "N" and the "threshold", to eliminate the error streaks. In Fig. 11 we can see the effects of "N" and the "threshold" on both the error streaks and the quality of the picture. Small values of "N" and low thresholds eliminate the error streaks but they tend to degrade the picture by eliminating thin objects such as the bottom of my glasses as shown in Fig. 11e. Large values of "N" and high thresholds leave some error streaks undetected and parts of others uncorrected but they do not degrade the picture. It is our opinion that Fig. 11c represents the best compromise between the degree of error correction necessary and the amount of degradation that can be tolerated in the picture.

Conclusion

Delta modulators can be used to encode video signals. The pictures that result from these signals are similar in quality to pictures that result from PCM encoders. The problem of channel errors causing "picture streaks" in delta modulated encoded video signals has been minimized by the algorithms presented in this paper.

References

1. Cuttler, "Delayed Encoding : Stabilizer for Adaptive Coders", IEEE Transactions on Communications. Vol. COM-19, Dec. 1971, pp.893-906.
2. C. L. Song, J. Garodnick, D. L. Schilling, "A Variable Step size Robust Delta Modulator", IEEE Transactions on Communications. Vol. COM-19, Dec. 1971, pp. 1033-1099.
3. D. J. Connor, "Techniques for Reducing the Visibility of Transmission Errors in Digitally Encoded Video Signals", IEEE Transactions on Communications, Vol. COM-21, June 1973, pp. 695-706.

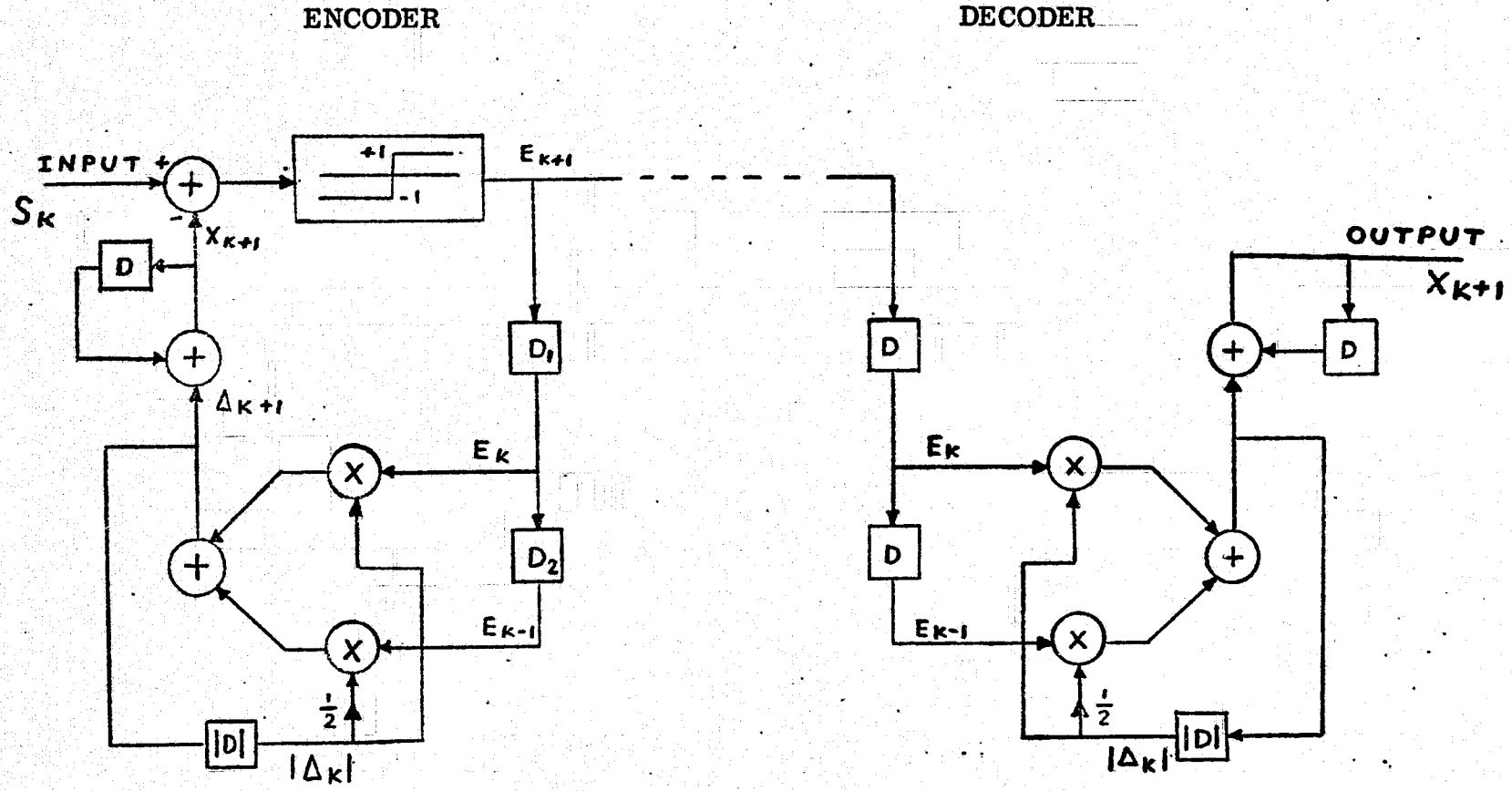


FIG. 1 DIGITAL ADAPTIVE DELTA MODULATOR

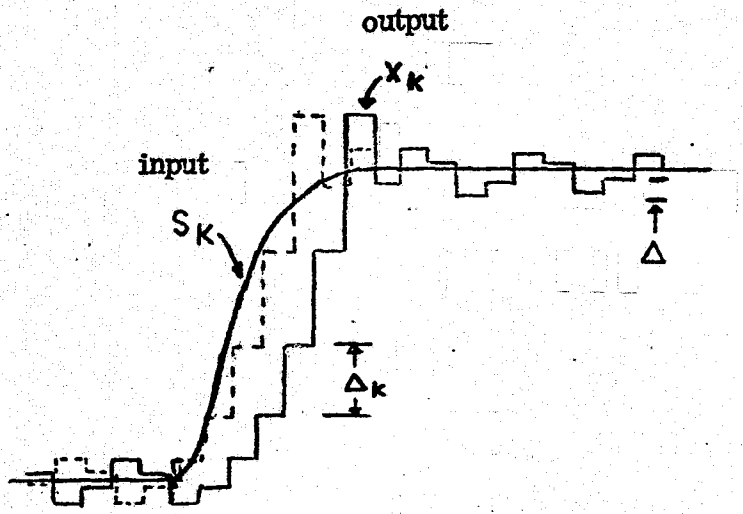


Fig. 2 Response of the Delta Modulator to a Step Input

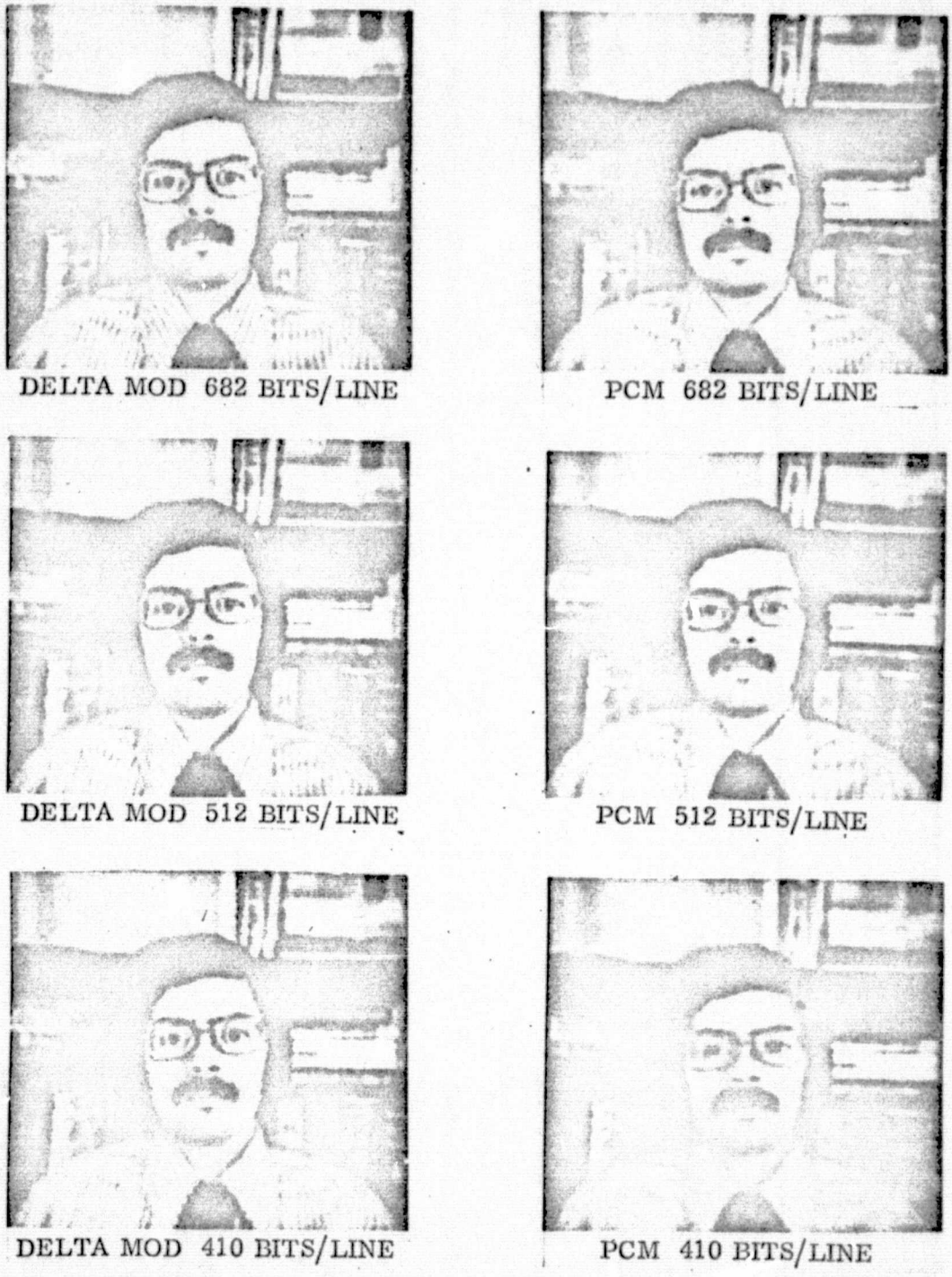


Fig. 3 A comparison of an adaptive delta modulator and a PCM encoder of video signals; 170 scaning lines per picture; PCM pictures have 64 quantization levels.

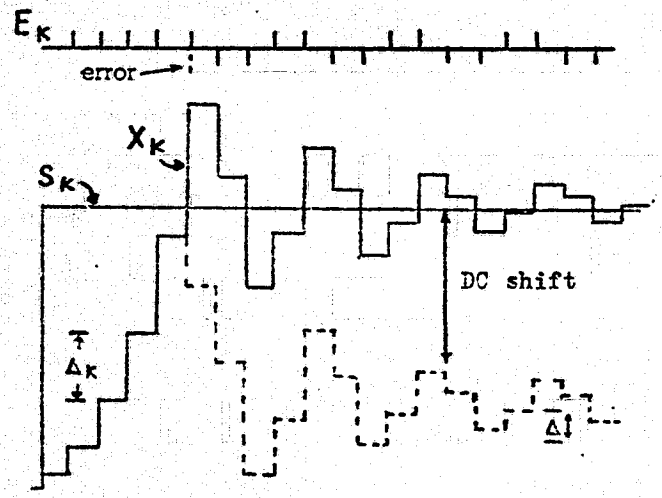


Fig. A No change in step size

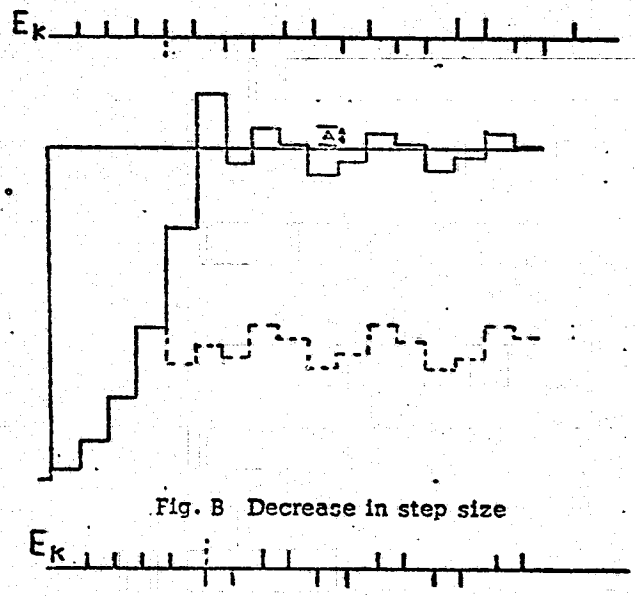


Fig. B Decrease in step size

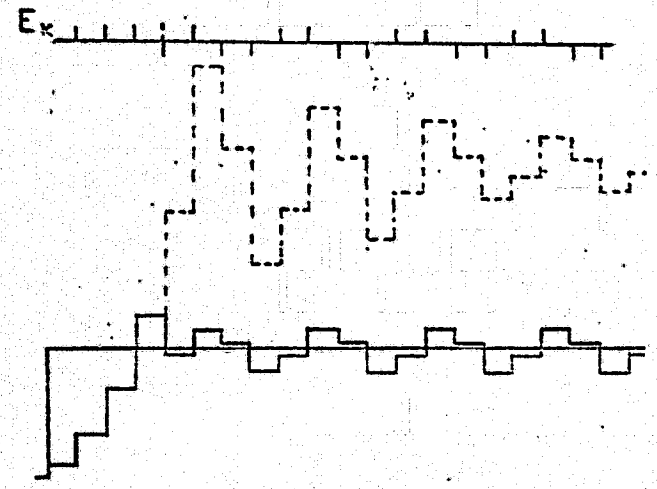
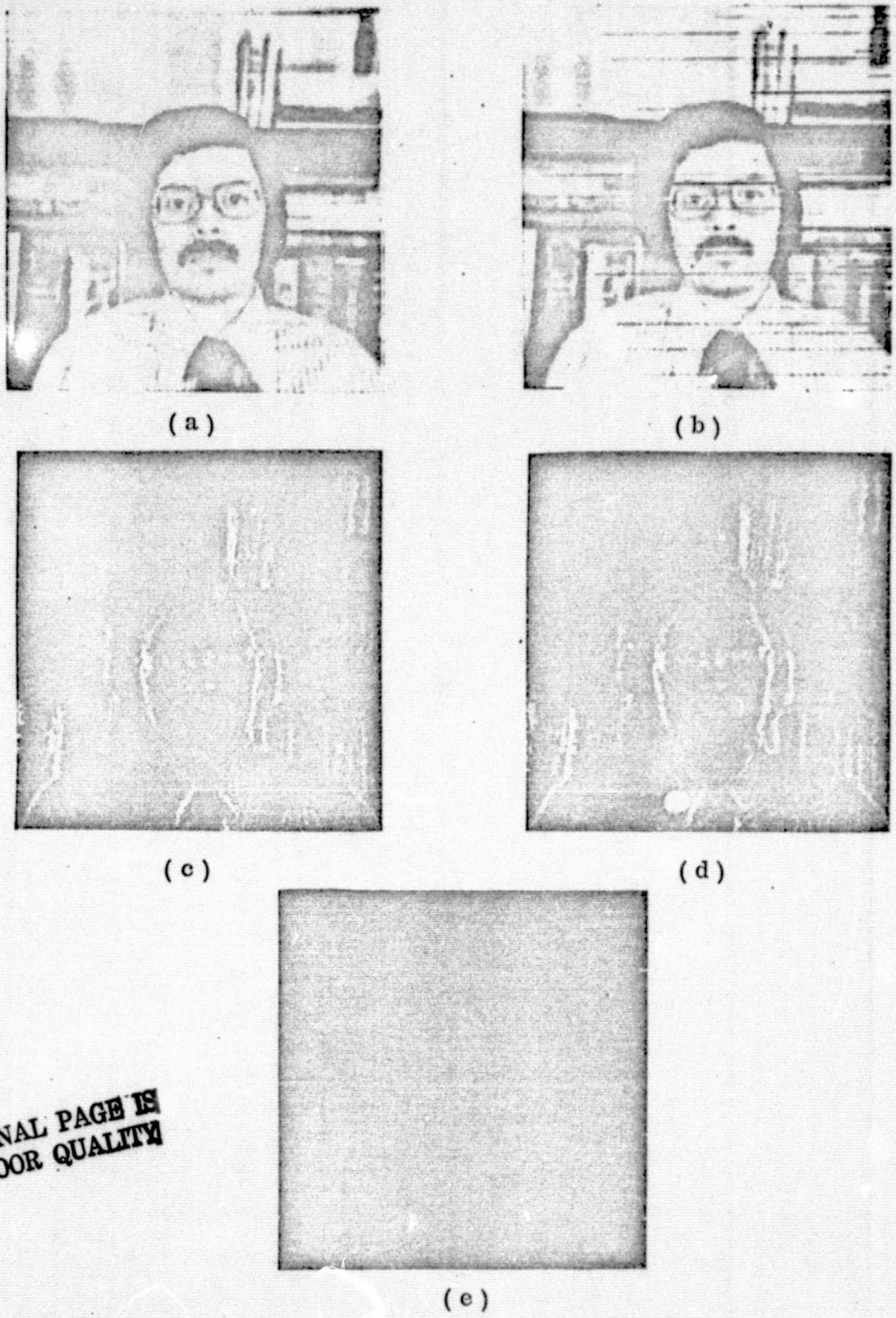


Fig. C Increase in step size

Fig. 4 The effects of a single channel error on the output of the delta modulator



(a) (b)

(c) (d)

(e)

Fig. 5 The effects of channel errors on delta modulated encoded pictures. (a) Delta modulated picture without channel errors; 682 bits/line, 170 lines. (b) Error rate of 4×10^{-4}. (c) Aboslute value of Δ_k (d) Absolute value of Δ_k with channel errors (e) Difference between (c) and (d), i.e. step size errors.

(a) Nosiy picture; 3×10^{-4} error rate; 1024 bits/line; 170 lines.

(b) 4 corrections per line; 1.5% increase in bit rate.

(c) 8 corrections per line; 3% increase in bit rate.

(d) 16 corrections per line; 6% increase in bit rate.

Fig. 6 The effect of sending the encoders estimate to the decoder to acheive error correction.

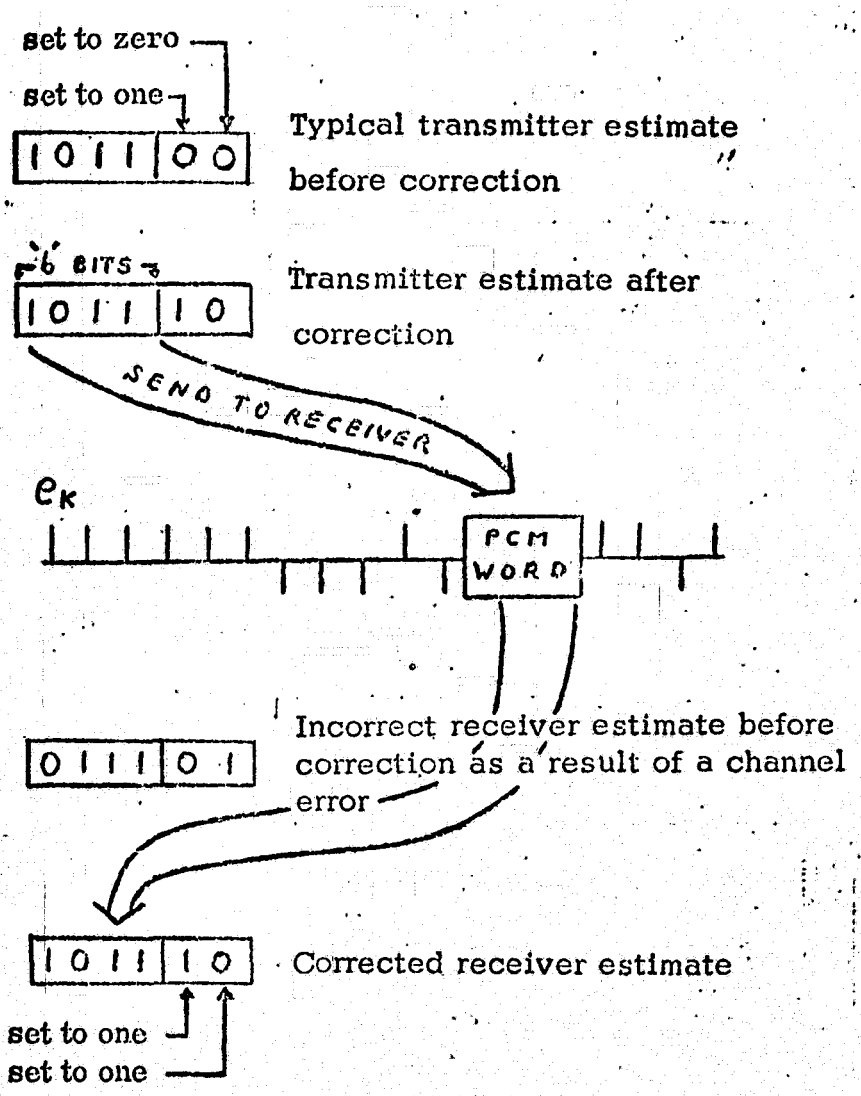
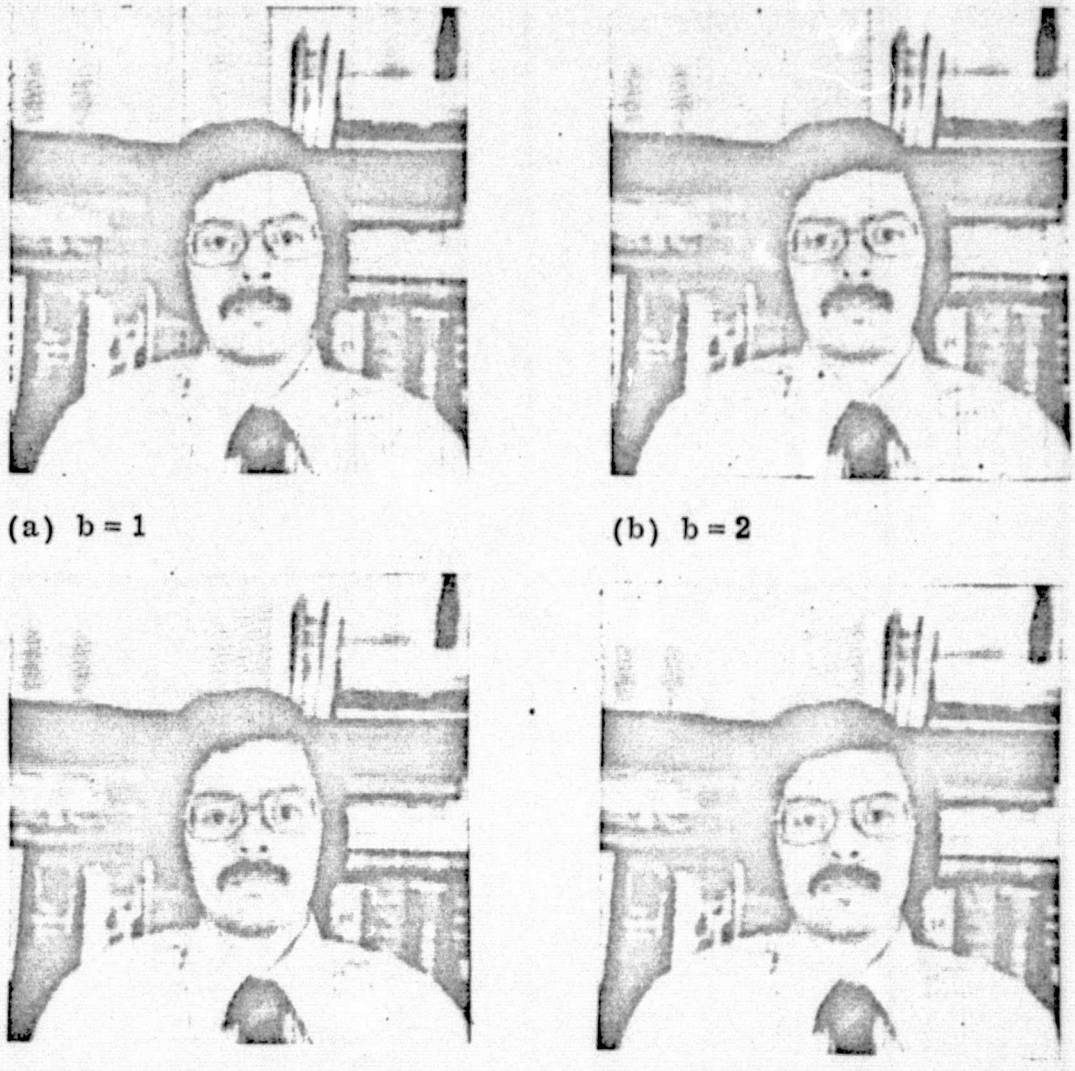


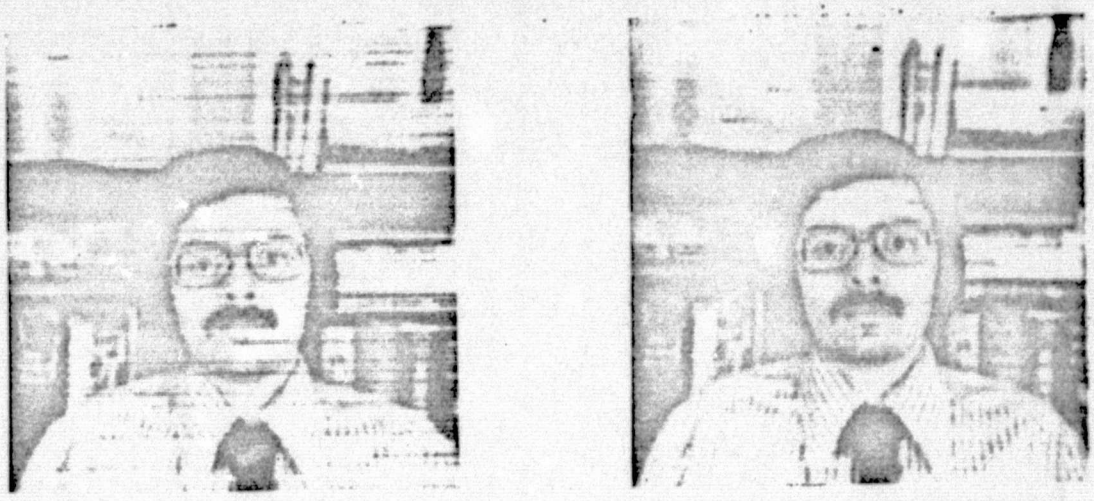
Fig. 7 Error correction is achieved by sending the "b" most significent bits of the transmitter's estimate to the receiver.



(a) b = 1 (b) b = 2

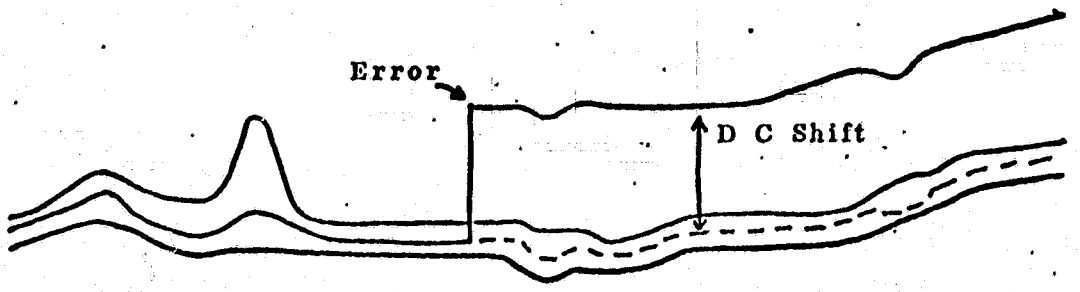
(c) b = 3 (d) b = 4

Fig. 8 The effect of sending only the "b" most significent bits of the encoders estimate to the decoder for error correction.

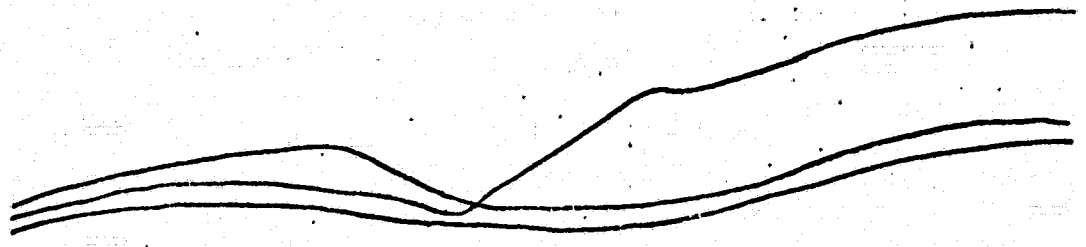


(a) (b)

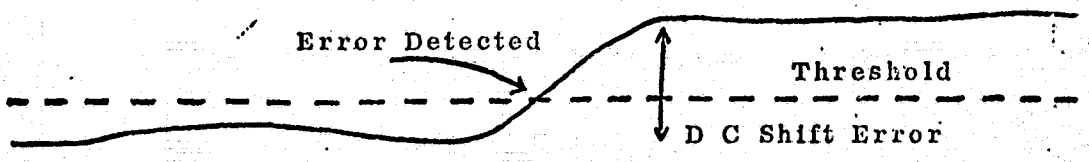
Fig. 9 Error correction is achieved by the use of a leaky integrator.
(a) Error rate 4×10^{-3}; 682 bits / line; 170 lines per picture
(b) Leaky integrator with a leak factor of 1 = .937



Part A. The Video Signal From 3 Consecutive Scanning Lines



Part B. The Video Signal After Averaging 'N' Consecutive Pixels.



Part C. The Absolute Value of the Difference Between the Middle Wave form and the Average of the Top and Bottom of Part B. is Shown in Part C.

Part D. The D C Shift of Part C. is Subtracted from the Video Signal to Produce the Corrected Signal of Part D.

Fig. 10 A line to line correlation algorithm for error correction

ORIGINAL PICTURE

(a)

NOISY PICTURE

(b)

THRESHOLD 1/16 P-P SIGNAL
AVERAGED OVER 1/16 OF A LINE

(c)

THRESHOLD 1/32 P-P SIGNAL
AVERAGED OVER 1/16 OF A LINE

(d)

THRESHOLD 1/16 P-P SIGNAL
AVERAGED OVER 1/128 OF A LINE

(e)

THRESHOLD 1/32 P-P SIGNAL
AVERAGED OVER 1/128 OF A LINE

(f)

Fig. 11 The results of using the line to line correlation algorithm on noisy pictures. 1000 bits / line; 170 lines per picture; error rate of 3×10^{-3}

Part II

Conversion From Pulse Code Modulation to Delta Modulation

A. Introduction

Frequently we encounter a situation where we have available a signal encoded in Pulse Code Modulation (PCM) format but we need the same signal in Delta Modulation (DM) form to be compatable with our overall communication system. Since both encoding techniques are digital in nature we would like to construct a PCM to DM converter which can easily be built with digital hardware. In addition, we want to be able to analytically evaluate its perform by calculating the resulting Signal-to-Noise Ratio (SNR).

In order to understand the problem that exists in a PCM to DM converter we introduce an information source, x(t), bandlimited to f_m. The PCM samples will then be available at the Nyquist rate, $2f_m$. In general, our digital DM operates at a rate, f_s, which is several times greater than the Nyquist rate. In Fig. 1 we show a basic digital DM which is described by the following set of equations:

$$e_x(k) = \text{sgn}[\xi(k)] \qquad (1a)$$

$$\xi_x(k) = x(k) - \hat{x}(k) \qquad (1b)$$

and

$$\hat{x}(k) = \hat{x}(k-1) + S_x(k) \qquad (1c)$$

where $S_x(k)$ = step size at the k^{th} interval.

Since we will primarily deal with conversion of voice signals, we shall specify the step size algorithm that is used with audio signals,

$$S_x(k) = |S_x(k-1)|\, e_x(k-1) + S e_x(k-2) \qquad (2)$$

where $S \equiv$ magnitude of the minimum step size.

Equation (2) is obtained from the general Song DM Algorithm* derived by minimizing a mean squared cost function.

If we use the above step size algorithm along with Eq. (1c), we can construct a DM estimate tree. A DM estimate tree is a graph of all possible paths that $\hat{x}(k)$ may follow starting with a set of initial conditions. The paths are generated from all possible binary sequences of $e_x(k)$. In Fig. 2 we show part of a DM estimate tree for the step size algorithm given in Eq. (2) on which we have superimposed two PCM samples, at points A and B. It is now easy to see where the difficulty lies in converting from PCM to DM. That is, in going from point A to points on the DM estimate tree adjacent to B, i.e., points B_1 and B_2, we can traverse any of four possible paths: l_1, l_2, l_3 or l_4. Which path to choose is exactly the problem that exists in PCM to DM conversion.

B. Linear Estimation Technique

The most straight forward approach to solve the multipath problem is to estimate the information signal between PCM samples and use these signal estimates as the input to an adaptive DM. We will then automatically generate a DM encoded signal, $e_x(k)$, and at the same time choose a path thru the DM estimate tree. If the Nyquist sampling period is $T = 1/2f_m$ and the DM sampling period is $\tau = 1/f_s$, then we shall require that T be an integer multiple of τ and therefore we must specify R PCM signal estimates every Nyquist period, where

$$R = T/\tau. \qquad (3)$$

* C. L. Song, G. Garodnick and D. L. Schilling, "A Variable Step Size Robust Delta Modulator", IEEE Trans. Commun. Tech., vol. COM-19, pp. 1033-1044, Dec. 1971.

A linear estimation technique that will accomplish this and can easily be implimented with digital hardware is shown in Fig. 3. Here we form a Differential PCM (DPCM) signal,

$$y(Rk) = x(Rk) - x(R(k-1)), \qquad (4)$$

which is then digitally scaled by a factor $1/R$, sampled and accumulated at the DM rate, f_s. The resulting signal, v(k), is then DM encoded and our conversion is complete. If we assume that the DM is operating fast enough such that there is minimum degradation between the DM estimate and its input, v(k), then we can approximate the estimate, denoted by w(t), as a piecewise linear curve obtained by connecting the PCM points. This will be the basis of our performance analysis and the calculation of in-band SNR.

C. Optimization of Converter Performance

The figure of merit for our converter will be the output in-band SNR. However, if we use the difference between x(t) and w(t) as the source of the output noise this will yield misleading results. Without loss of generality, we can amplify and delay w(t) to improve SNR. This is because attenuation and phase shift cause no distortion in audio signals. In general, if w(t) were really obtained from a linear transformation of the PCM samples, then we could conceivably use a linear filter to perform the inverse transformation and have no signal degradation at all. However, since w(t) is really obtained via a non-linear transformation, i.e., thru the DM, the inverse transformation filter may not be physically realizable.

If we allow w(t) to be scaled by a factor, K, and delayed by a time, γ,

then the error signal that we are concerned with becomes

$$d(t) = x(t) - K\, w(t-\gamma), \tag{5}$$

We can now determine the autocorrelation function of the error signal as

$$R_d(\tau) \equiv E[\, d(t)\, d(t+\tau)\,] \tag{6}$$

$$R_d(\tau) = R_x(\tau) + K^2 R_w(\tau) - K[\, R_{xw}(\tau-\gamma) + R_{xw}(\tau+\gamma)\,], \tag{7}$$

where

$R_x(\tau)$ = the autocorrelation function of x(t),

$R_w(\tau)$ = the autocorrelation function of w(t)

and

$R_{xw}(\tau)$ = the crosscorrelation function of x(t) and w(t).

Taking the Fourier transform of Eq. (6), we obtain the power spectral density of the error signal,

$$G_d(f) = G_x(f) + K^2 G_w(f) - 2K\cos(2\pi f\gamma)\, G_{xw}(f). \tag{8}$$

Finally, integrating over the input signal frequency band, we can find the in-band noise power, P_d, and the in-band signal power, P_s, where

$$P_d \equiv \int_{-f_m}^{f_m} G_d(f)\, df \tag{9}$$

$$P_d = P_x + K^2 P_w - 2K\int_{-f_m}^{f_m} \cos(2\pi f\gamma)\, G_{xw}(f)\, df \tag{10}$$

and

$$P_s = K^2 P_w. \tag{11}$$

Using Eqs.(10) and (11), we can form our figure of merit as

$$Q \equiv SNR \equiv P_s/P_d, \tag{12}$$

which is found to be

$$Q = \frac{K^2 P_w}{P_x + K^2 P_w - 2K\mu(\gamma)} , \qquad (13)$$

where

$$\mu(\gamma) \equiv \int_{-f_m}^{f_m} \cos(2\pi f \gamma)\, G_{xw}(f)\, df . \qquad (14)$$

Now we seek to maximize Q by applying

$$\frac{\partial^2 Q(K, \gamma)}{\partial \gamma\, \partial K} = 0 . \qquad (15)$$

If we set $\partial Q/\partial \gamma$ equal to zero, we find that γ is a constant, denoted as γ_o, independent of K, and satisfying the relationship

$$\int_{o}^{f_m} f \sin(2\pi f \gamma_o)\, G_{xw}(f)\, df = 0. \qquad (16)$$

The result of equating $\partial Q(K, \gamma_o)/\partial K$ to zero gives us the optimum K as

$$K_o = P_x/\mu(\gamma_o). \qquad (17)$$

Finally we have the maximum SNR, $Q_o = Q(K_o, \gamma_o)$, expressable as

$$Q_o = \frac{P_x P_w}{P_x P_w - \mu^2(\gamma_o)} , \qquad (18)$$

where

$$\mu(\gamma_o) = \int_{-f_m}^{f_m} \cos(2\pi f \gamma_o)\, G_{xw}(f)\, df. \qquad (18a)$$

D. Evaluation of Converter SNR

In order to determine the SNR of our converter, we must first calculate the autocorrelation function of w(t) and the crosscorrelation function of x(t) and w(t). To accomplish this we set up a coordinate system depicted in

Fig. 4 and observe that the piecewise linear curve, w(t), is actually the sum of a sequence of ramps of slope m_i and steps of amplitude w_i. To insure stationarity in the random process, w(t), we introduce a random starting time for the ramp functions shown as $-\tau_o$. We also introduce the random variable, λ, which is uniformly distributed in the interval 0 to T. To completely define our system, we specify the parameters

$$j = [\tau/T] \equiv \text{the greatest integer} \leq \tau/T,$$

$$\tau = jT + \eta_o,$$

and

$$\eta = \tau - jT + \lambda, \text{ for } 0 \leq \lambda \leq T.$$

We will first determine the autocorrelation function of w(t). In order to facilitate matters let us express the ramps at times t_1 and t_2 as

$$z(t_1) = m_o\lambda \tag{19}$$

and

$$z(t_2 = t_1 + \tau) = \begin{cases} m_j\eta, & \eta \leq T \\ m_{j+1}(\eta - T), & \eta > T. \end{cases} \tag{20}$$

Now we can find the autocorrelation function of the ramps, i.e.,

$$\begin{aligned} E[\, z(t_1)\, z(t_2)\,] = {} & (1/2T)\eta_o(T-\eta_o)^2 R_m(j) + (1/3T)(T-\eta_o)^3 R_m(j) \\ & + (1/2T)(\eta_o - T)[T^2 - (T-\eta_o)^2]\, R_m(j) \\ & + (1/3T)[T^3 - (T-\eta_o)^3]\, R_m(j+1), \end{aligned} \tag{21}$$

where

$$R_m(j) = E(m_o m_j) = [\, 2R_x(j) - R_x(j+1) - R_x(j-1)\,]/T^2 \tag{22}$$

and

$$R_x(j) \equiv R_x(jT).$$

If we combine the ramps and the steps, then we can represent w(t) at times t_1 and t_2 in the following way:

$$w(t_1) = w_o + m_o\lambda \qquad (23)$$

and

$$w(t_2) = w_j + m_j\eta \quad , \ \eta \le T$$
$$\qquad\quad w_{j+1} + m_{j+1}(\eta - T), \ \eta > T \qquad (24)$$

Using the result found in Eq. (21), we proceed to evalutate the autocorrelation function of w(t), that is,

$$\begin{aligned} E\,[w(t_1)w(t_2)\,] = {} & (1/T)(T-\eta_o)R_w(j) + (1/T)\,(T-\eta_o)^2(2T+\eta_o)R_m(j) \\ & + (1/2T)(T-\eta_o)^2\,E\,(m_o w_j) + (1/2T)(T^2-\eta_o{}^2)E(m_j w_o) \\ & + (\eta_o/T)R_w(j+1) + (\eta_o{}^2/6T)(3T-\eta_o)R_m(j+1) \\ & + (\eta_o/2T)(2T-\eta_o)\,E(m_o w_{j+1}) + (\eta_o/2T)\,E(m_{j+1}w_o), \quad (25) \end{aligned}$$

where

$$R_w(j) = E(w_o w_j) = R_x(j) \qquad (26)$$

and

$E\,(m_i w_k)$ is a function of $R_x(j)$.

Simplifying Eq. (25) we arrive at the final form of the autocorrelation function of w(t)

$$\begin{aligned} R_w(\tau) = {} & (\eta_o{}^3/6T^3)\,[\,3R_x(j) + R_x(j+2)\,] \\ & + [\,(T-\eta_o)^3/6T^3\,]\,[\,R_x(j-1) + 3R_x(j+1)\,] \\ & - (1/T^2)\,[\,\eta_o{}^2R_x(j) + (T-\eta_o{}^2)\,R_x(j+1)\,] \\ & + (2/3)\,[\,R_x(j) + R_x(j+1)\,] \end{aligned} \qquad (27)$$

where

$$0 \le \eta_o \le T$$

and recall that

$$j = [\,\tau/T\,] = \text{the integer} \le \tau/T.$$

Employing the notation for $w(t_1)$ introduced in Eq. (23), it becomes quite a less formidable task to calculate the crosscorrelation function between x(t) and w(t). This can be expressed as

$$R_{xw}(\tau) = R_{wx}(\tau) = E\left[w(t_1)\, x(t_1+\tau) \right]$$

$$= (1-\lambda/T)\, R_x(\tau+\lambda) + (\lambda/T)\, R_x(\tau+\lambda-T), \qquad (28)$$

where the average in Eq. (28) is over the random variable λ. This result can be further reduced to

$$R_{xw}(\tau) = (1/T)\int_0^T (1-\lambda/T)\left[R_x(\tau+\lambda) + R_x(\tau-\lambda) \right] d\lambda. \qquad (29)$$

At this point we see that if the correlation function of x(t) is specified, then we can formulate the correlation function of d(t) as given in Eq. (7) and its power spectral density from Eq. (8). Furthermore, we can continue to simplify our results and show that the power spectral density obtained from $R_{xw}(\tau)$ is a function of the power spectral density of x(t). Taking the Fourier transform of Eq. (29), we obtain, in its simplest form,

$$G_{xw}(f) = \left[2(1-\cos(2\pi fT))/(2\pi fT)^2 \right] G_x(f). \qquad (30)$$

We are currently in the process of determining numerical results for input signals having various spectral shapes. Among those being investigated are signals with flat and triangular power spectral densities up to f_m. In order to obtain these numerical results we must determine not only the various power spectral denities and in-band powers discussed above, but we must also solve the optimization relationship given in Eq. (16). Then we can evaluate the maximum SNR, Q_o, as given by Eq. (18).

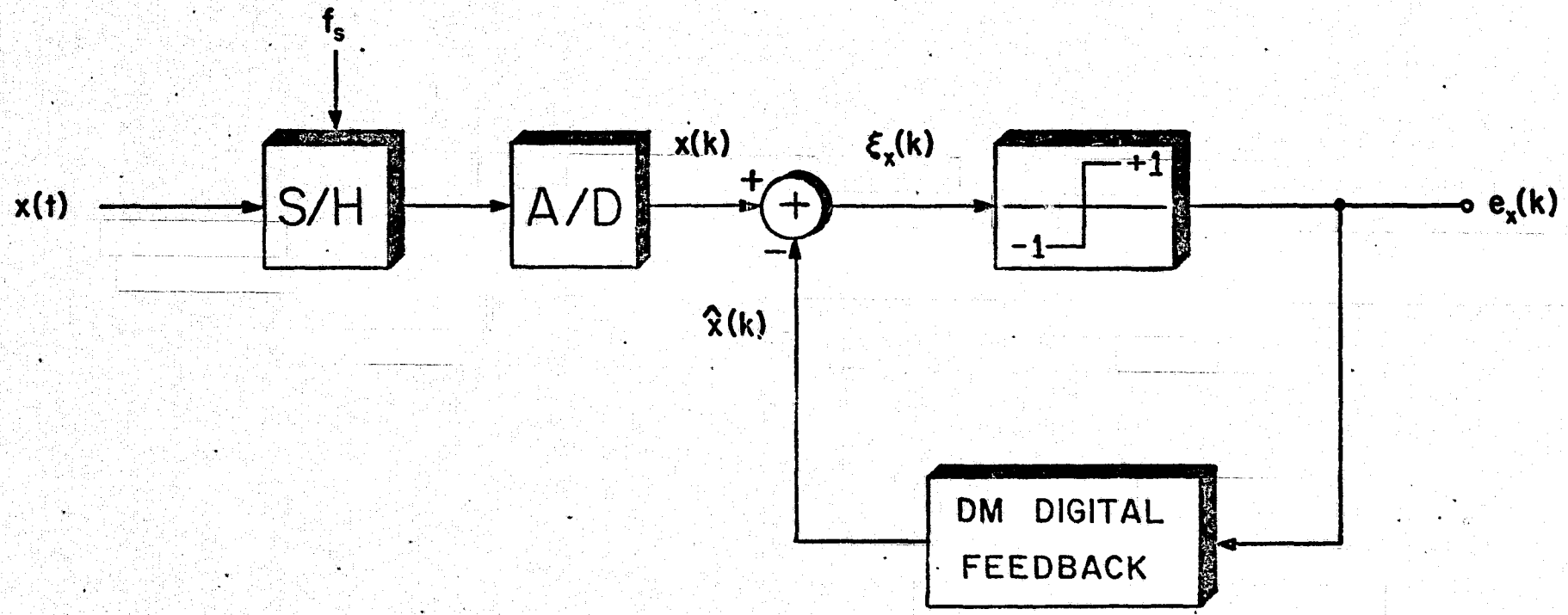


Fig. 1. General Form of an All-Digital DM

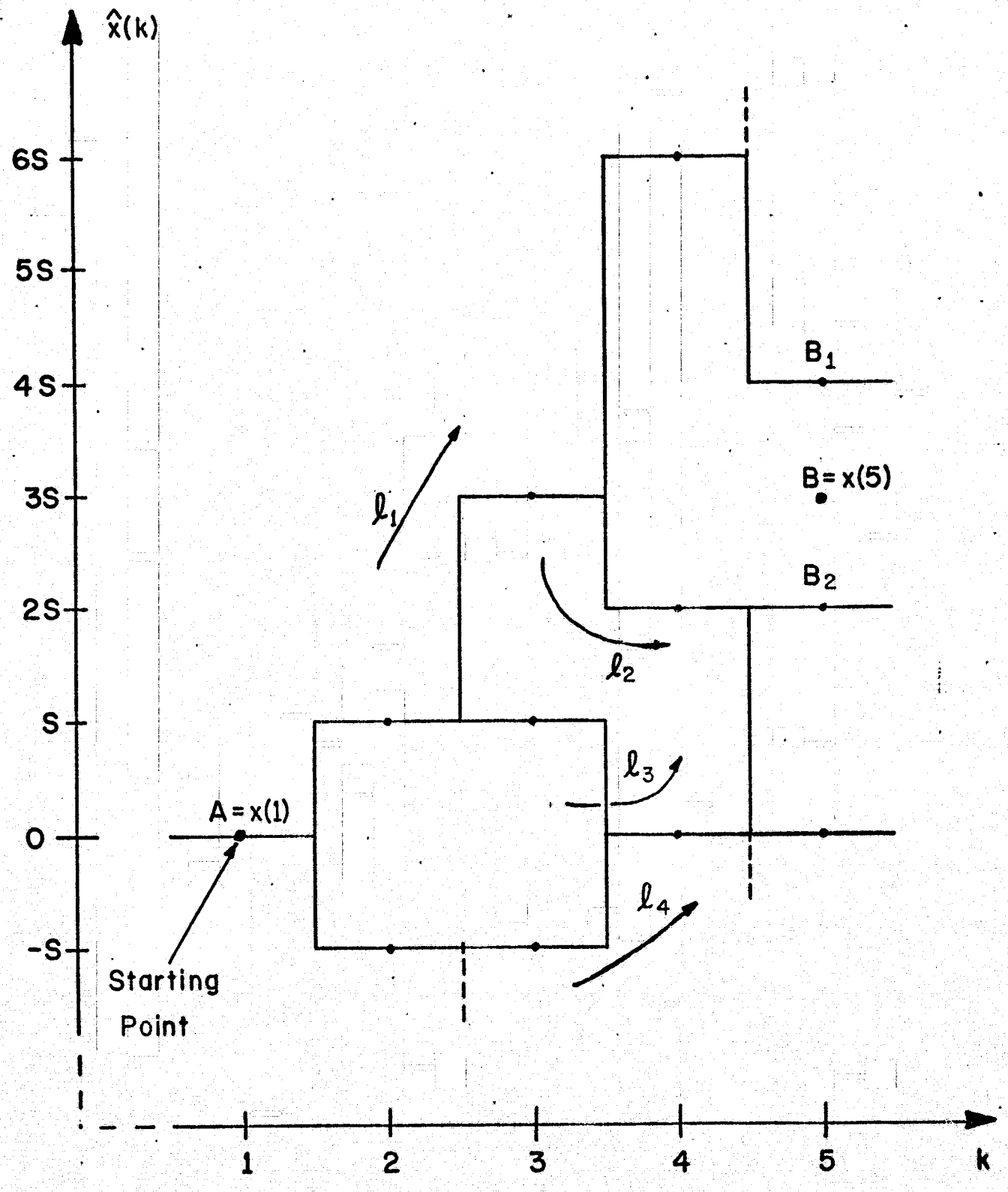


Figure 2. Possible DM Signal Estimate Paths

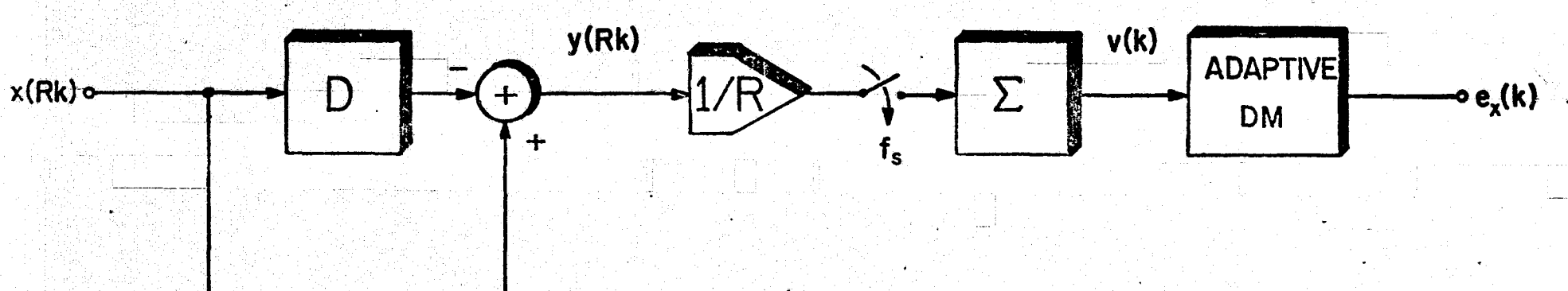


Fig. 3. PCM to DM Converter

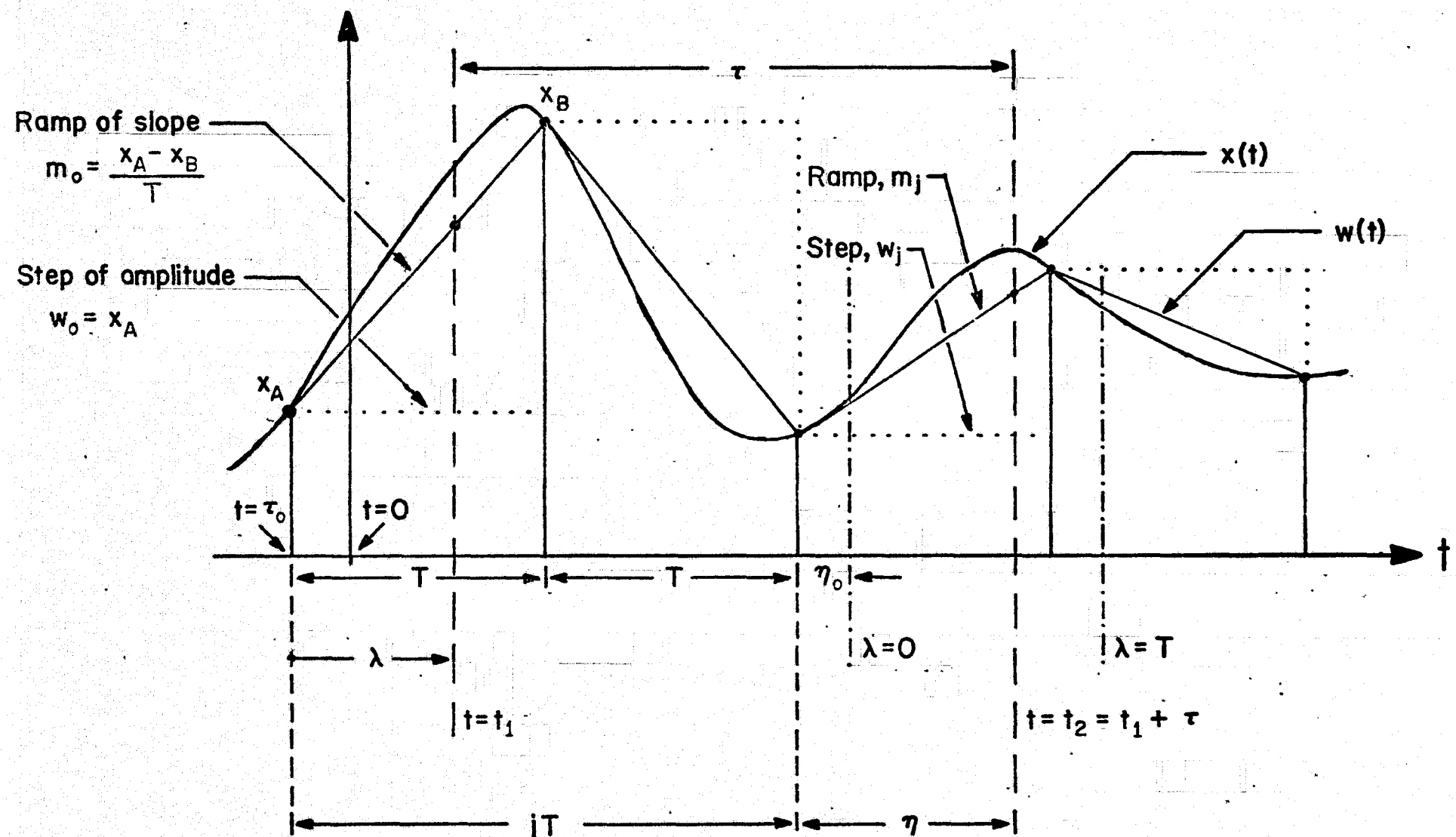


Fig. 4. Coordinate System for Converter SNR Analysis

Part III A PLL Using a Nonlinear Processor Instead of a Loop Filter

I. Introduction

The Phase Locked Loop (PLL) is a device which has wide application in the field of communications. It can be used for carrier tracking, FM demodulation, and bit synchonization. An analog PLL is shown in Fig. 1 and consists of a Phase Detector (PD) or Comparator (PC), a linear filter and a Voltage Controlled Oscillator (VCO). The incoming signal is of the form.

$$f(t) = -2 \cos \left[2\pi f_c t + \phi_m(t)\right] + n_w(t)$$

and the VCO waveform is

$$v(t) = \sin \left[2\pi f_o t + \phi_v(t)\right]$$

where f_c is the nominal input carrier frequency, f_o is the VCO carrier frequency, $n_w(t)$ is white Gaussian noise, ϕ_m and ϕ_v are the phase of the input and VCO waveform respectively. In FM demodulation f_o is usually close to f_c and $\phi_v(t)$ is then an approximation of $\phi_m(t)$ the information signal. In carrier tracking or bit synchonization f_o is not in general equal to f_c. Instead during the transient there is a difference between the two, $\Delta f = f_1 - f_2$, and the loop attempts to minimize this difference. When Δf is "small" the loop is said to be in lock. The time required to do this is the acquisition time. Obviously the shorter this time the better.

Most PLL's whether analog, hybrid or digital in nature use linear filters in the loop as the processor. The PD extracts the sine of the difference between the arguments of the input and VCO waveform. This signal is then filtered and used to drive the VCO in such a way as to reduce the phase error between the input signal and VCO waveform.

In the system we are considering however the filter is not linear. Instead the filter we use consists of a hard limiter and processor (Fig. 2) which steps the VCO in discrete steps towards nulling the phase error $\phi_e = \phi_m - \phi_v$.

This affords us greater control over the VCO and lends itself to a discrete analysis of the systems.

Two systems we will consider use a linear processor and an adaptive processor. The two processors will resemble Delta Modulators (DM) in that the algorithm which produces the input to the VCO is based on the algorithms of DM's we have been investigating. The adaptive algorithm has been found to be useful in video application since it has a rapid rise time. This will allow faster acquisition in carrier tracking problems.

Let us now examine the system we have proposed.

II. The Basic Systems

In Fig. 2 if we replace the adaptive processor by a unity gain we have a linear processor.

Consider the system shown in Fig. 3.

where $$s(t) = \sin [w_c(t) + \phi_m(t)] + n_w(t) \qquad (1a)$$

$$f(t) = \sin \phi_e(t) + n_1(t) \qquad (1b)$$

$$\phi_e(t) = \phi_m(t) - \phi_v(t) \qquad (1c)$$

and $n_1(t)$ is zero mean white Gaussian noise [1] of power spectral density $n_o/2$. $f(t)$ is the input to an integrate and dump filter which produces an output ($v_o(t)$) voltage every T seconds where

$$v_o(t) = y(t) + n_f(t). \qquad (2)$$

$y(t)$ is the output of the integrate and dump filter due to the deterministic signal and $n_f(t)$ is the component due to the noise, i.e.

$$y(t) = \frac{1}{T} \int_o^T \sin \phi_e(\lambda)\, d\lambda \qquad (3a)$$

$$n_f(t) = \frac{1}{T} \int_o^T n_1(\lambda)\, d\lambda \qquad (3b)$$

Let us now examine the noise term $n_f(t)$. Since $n_1(t)$ is zero mean white Gaussian noise $n_f(t)$ is represented by a Wiener Levy process [2].

$$\overline{n_f(t)} = 0 \qquad (4a)$$

and

$$\text{Var}\, n_f(t) = \frac{n_o}{2T} = \sigma^2 \qquad (4b)$$

We therefore have that the probability density function (pdf) of the noise component is

$$P_{n_f}(\beta) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\beta/2\sigma^2} \qquad (5a)$$

where σ^2 is given above.

From Fig. 1, one can see that

$$v_o'(t) = \frac{\dot{\phi}_v(t)}{G_{vco}} = \text{sgn}\, v_o(t) \qquad (6a)$$

or

$$\dot{\phi}_e(t) = \dot{\phi}_m(t) - G_{vco}\, \text{sgn}\, v_o(t) \qquad (6b)$$

Let us approximate the differential $\dot{\phi}_e$ by a difference as follows:

$$\dot{\phi}_e(t) = \left\{\phi_e\left[(k+1)T\right] - \phi_e\left[kT\right]\right\} / \tau \qquad (7)$$

so we have

$$\phi_e\left[(k+1)T\right] - \phi_e\left[kT\right] = \dot{\phi}_m(kT) - G_o \text{sgn}\left[\frac{1}{T}\int_o^T \sin\phi_2(\lambda)d\lambda + n_f(k)\right] \qquad (8)$$

where $G_o = G_{vco}\,\tau$ and $kT \le \lambda \le (k+1)T$

We now consider the above problem for one case of input information signal, $\phi_m(t)$.

A. Phase Offset

1. Linear mode

Let $\phi_m(t)$ be a constant ϕ_f so that $\dot{\phi}_m(t) = 0$ assume initially that $\phi_v(0) = \phi_o$ so that $\phi_e(0) = \phi_m(0) - \phi_v(0) = \phi_f - \phi_o = \Delta\phi$. The difference equation

governing the above is then

$$\phi_e(k+1) - \phi_e(k) = -G_o \operatorname{sgn}\left[v_o(k)\right] \qquad (9)$$

where we have normalized T to unity for simplicity.

From the above equation we notice the phase error, $\phi_e(k)$ will change from its previous value $\phi_e(k-1)$ by $\pm G_o$ depending upon the sign of $v_o(k-1)$. If we start with $\phi_v(o) = \phi_o$ and progress through the various states $\phi_v(k)$ can attain we find that a tree structure as indicated in Fig. 4 will be developed.

Since in a PLL, one wishes to drive the (steady state) error to zero, i.e. acquire lock we would like to know how long it would take to reach steady state. Let us now define what we mean by steady state. It is known that many DM's when responding to a step use will attain a repetitive pattern after some inital transient.

For a linear DM, which is the type illustrated in Fig. 1, the periodic pattern is a square wave as indicated in Fig. 5. For another type of, DM i.e. an adaptive DM, the pattern may appear as in Fig. 6, illustrated for the Song Video Mode DM (by inserting the proper processor after the sgn function in Fig. 2. this type pattern will emerge). As one can see from Fig. 4 there are many ways to approach steady state. We are interested in finding the longest or worst time case in which we are assured to within a Δ (delta) region of acquiring steady state lock (i.e. a confidence interval). To do this we will proceed as follows : we will find the transition probabilities between adjacent states; then find the path probabilities for each path that reaches a state which is representative of acquisition (e.g. in Fig. 4 states (28, 46, ...); (36, 56, ...)). We will then sum the probabilities of the most likely paths until this sum is within our confidence interval. The longest for this to happen i.e. the worst case will be our acquisition time. We may then say for example that we are 95% sure that in 50 sampling instants we will reach steady state.

In reference to the condition of a phase offset let

$$z(k) = y(k) + n_f(k)$$

Since $y(k) = \frac{1}{T}\int_0^T \sin \phi_e(\lambda)\,d\lambda$ it is a random variable at time k because $\phi_e(\lambda)$ is a random variable. Therefore

$$p_Z(\gamma/y = Y) = p_{n_f}(\gamma - Y / y = Y) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) e^{-(\gamma - y)^2/2\sigma^2} \qquad (10a)$$

and

$$F_3(\gamma / y = Y) = \text{Prob}(\gamma \leq \Gamma / y = Y) \qquad (10b)$$

where σ^2 is given above (Eq. (4b)). Now, let us define x(k) as follows:

$$x(k) \triangleq G_o \,\text{sgn}\,[z(k)] \qquad (11)$$

so that

$$p_{x(k)}(\alpha = +G_o) = p_{z(k)}(\gamma \geq 0) = 1 - F_{z(k)}(o) \qquad (12a)$$

$$p_{x(k)}(\alpha = -G_o) = p_{z(k)}(\gamma < 0) = F_{z(k)}(o) \qquad (12b)$$

and

$$p_{\phi_e}(\phi_e(k+1) / \phi_e(k)) = p_x(\alpha + \phi_e(k)) \qquad (13)$$

The procedure for finding the time to acquire is now clear. We use Eq. (13) in conjunction with Eqs. (12) and (10) to find the transition probabilities.

Since each transition is independent of the previous one (the process is a Markov - one process) the path probabilities are the product of the transition probabilities of each state pair in a given path. These path probabilities are summed until the total is greater than or equal to the confidence interval desired. We are presently involved in developing an algorithm which will simplify the computation of path probabilities and then allow us to evaluate the performance of the system and determine the optimum parameters.

2. Adaptive Mode

In the adaptive mode we make the following modification. The adaptive processor now takes on the form indicated by the equations below:

$$\phi_e(k+1) = \phi_e(k) + \Delta\phi_e(k) \qquad (14a)$$

$$\Delta\phi_e(k) = | \Delta\phi_e(k-1) | (b(k-1) + \tfrac{1}{2} b(k-2)) \qquad (14b)$$

These equations lead to an exponentially rising estimate which can be analyzed as before. The states are now no longer separated by a uniform amount but by a varying level (Fig. 7). A characteristic of this system is to have large overshoots, but an overshoot suppression algorithm which has been developed [3] for this mode of operation can be used to reduce the overshoot. This enables the system to reach steady state more rapidly and this will reduce acquisition time.

Presently computer simulation of this system are being investigated in reference to acquisition time.

References

(1) Viterbi, A. J. Principles of Coherent Communication, McGraw Hill, New York, 1966.

(2) Papoulis, A. Probability, Random Variable and Stochastic Processes, McGraw Hill, New York, 1965, pp. 436-437.

(3) Weiss, L. et.al. "Overshoot Suppression in Adaptive Delta Modulator Links for Video Transmission", National Telecommunications Conference, Atlanta, Georgia, Nov. 26-28, 1973.

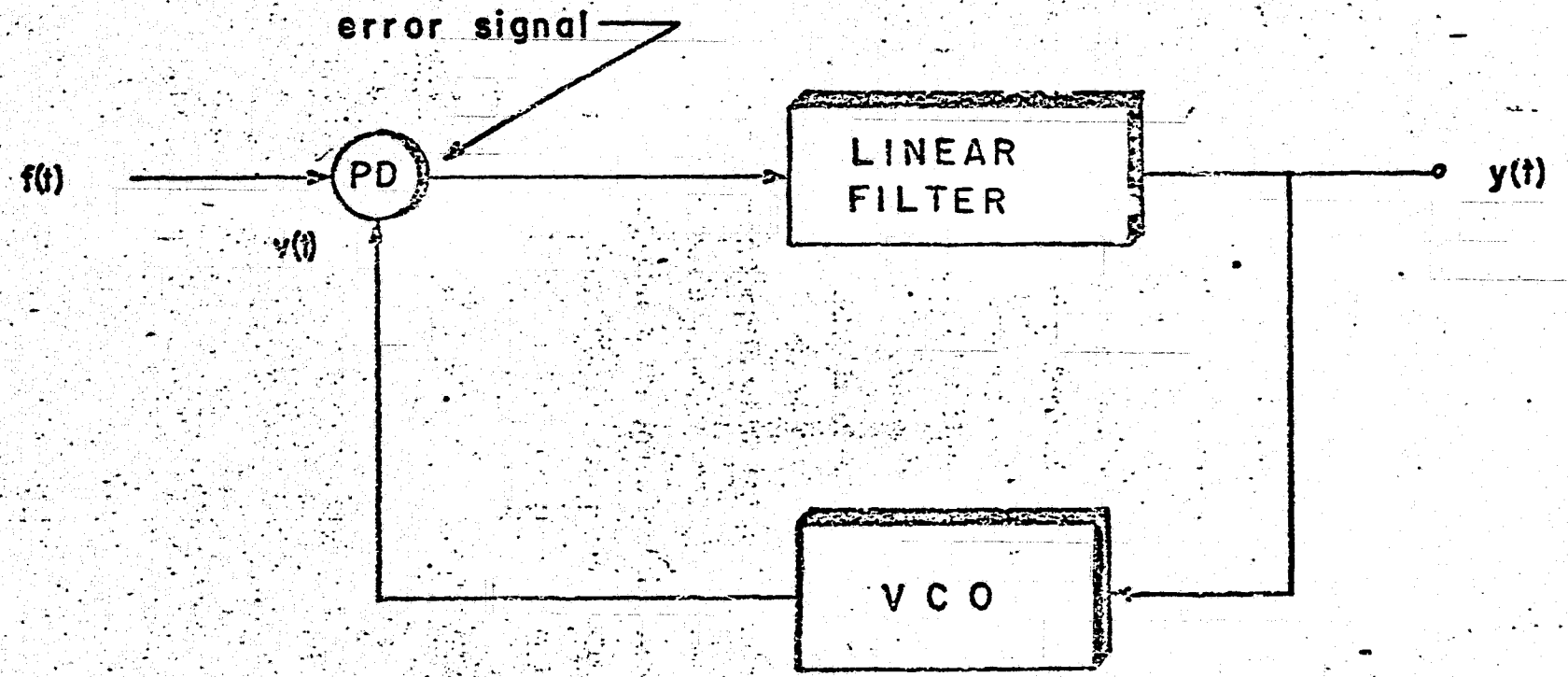


Figure 1. Basic PLL

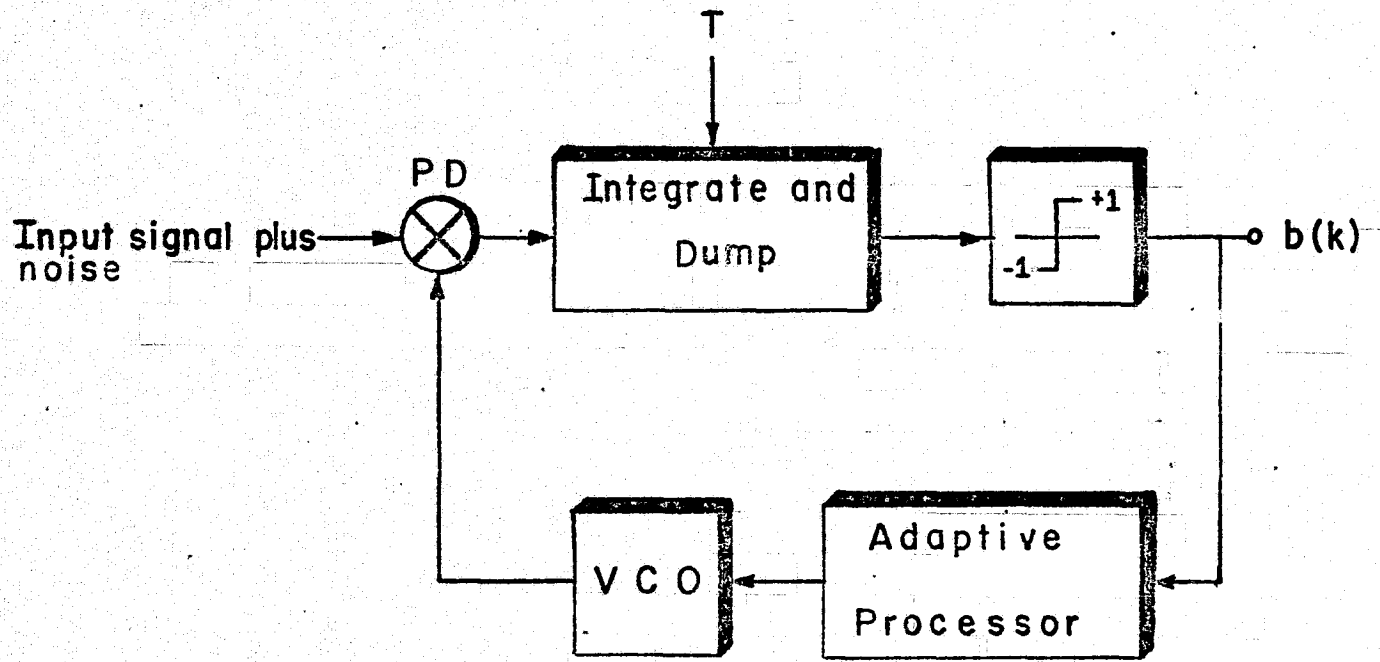


Figure 2. A General Adaptive PLL

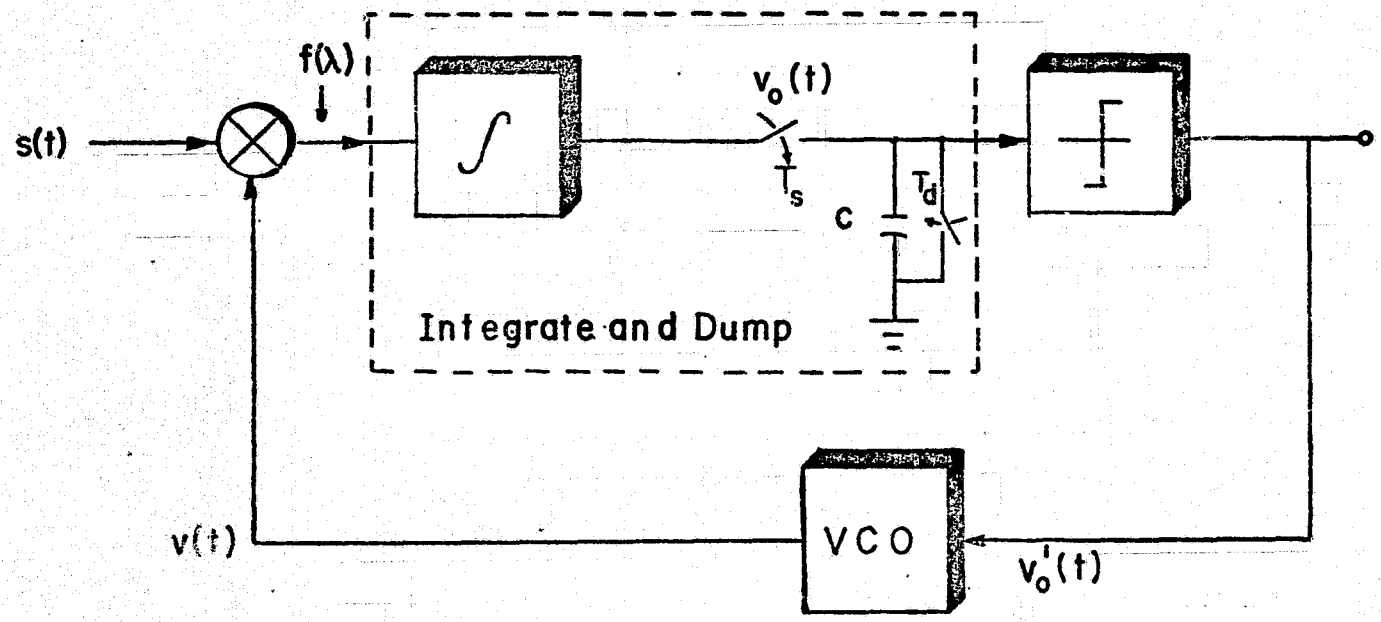


Figure 3. System with Linear Processor

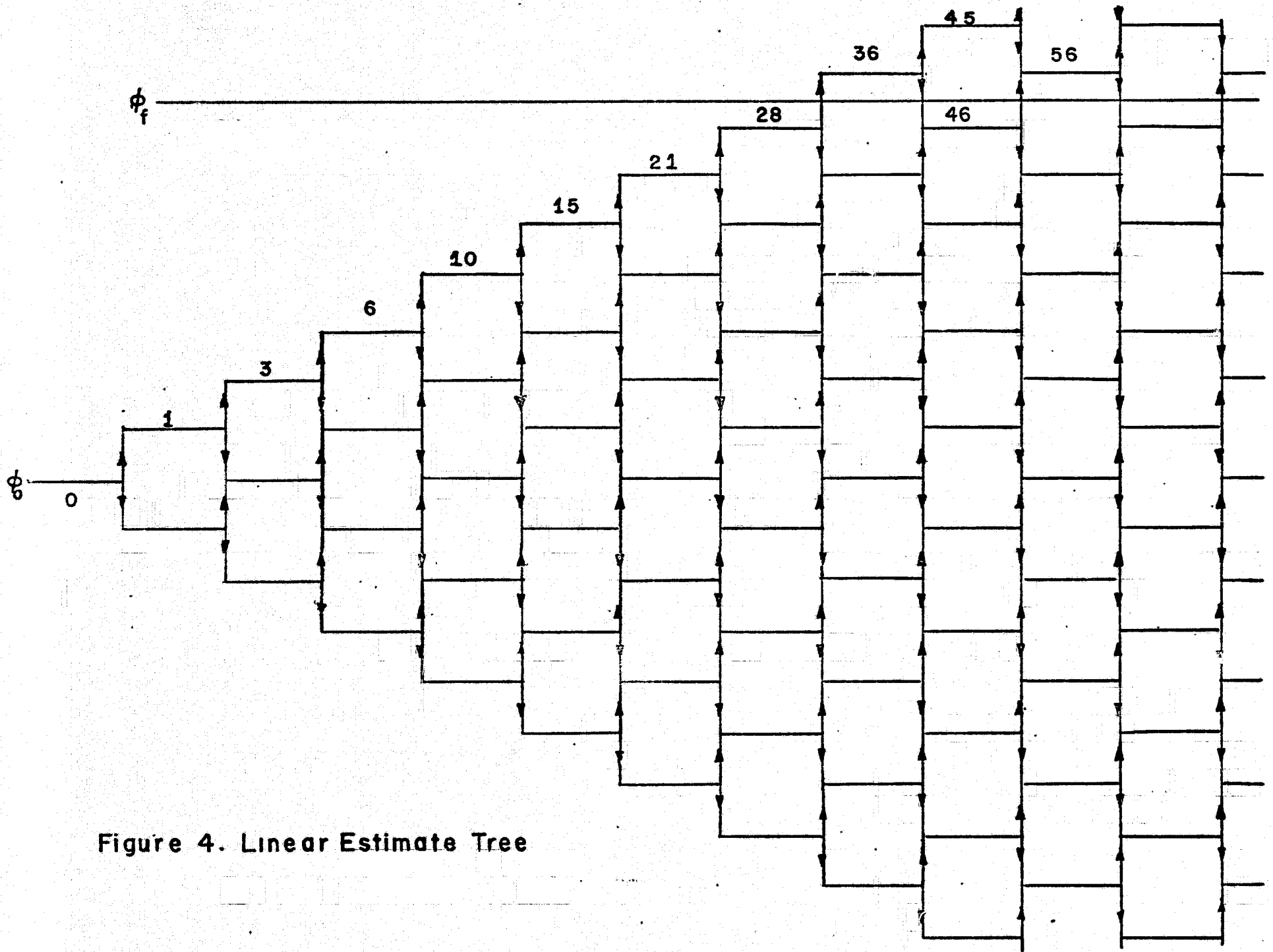


Figure 4. Linear Estimate Tree

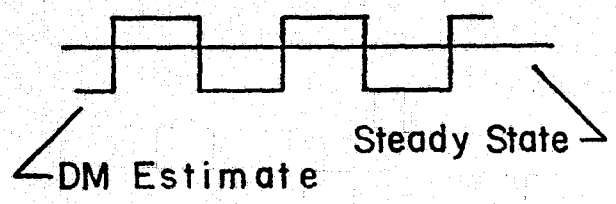


Figure 5. Linear DM Steady State Pattern

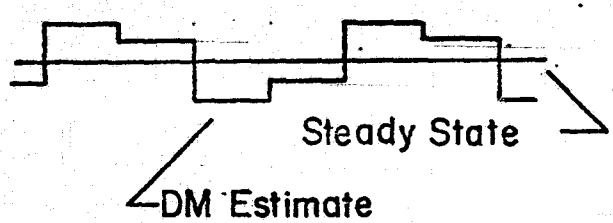


Figure 6. Adaptive DM Steady State Pattern

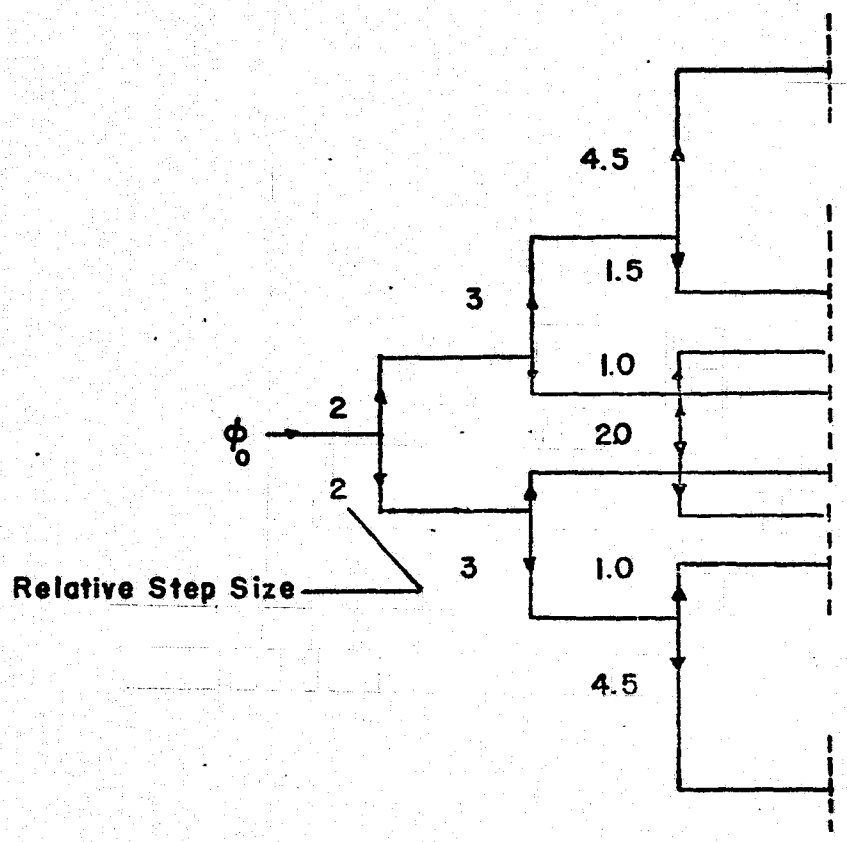


Figure 7. Adaptive Estimate Tree

Part IV Publications and Presentations

1) M. Z. Ali, I. Paz, N. R. Scheinberg and D. L. Schilling, "A Technique for Correcting Transmission Errors in Delta Modulation Channels", 1974 International Conf. on Commun. Record, pp. 44F-1 - 44F - 6.

2) J. L. LoCicero, D. L. Schilling and J. Garodnick, "Direct Arithmetic Processing of Delta Modulation Encoded Signals", 1974 National Telecommun. Conf. Record, pp. 392-397.

3) N. R. Scheinberg, D. L. Schilling, M. Z. Ali and I. Paz, "A Technique for Correcting Transmission Errors in Video Delta Modulation Channels", 1975 International Conf. on Commun. Record, pp. 27-21-27-25.

4) L. Weiss, I. Paz and D. L. Schilling, "Video Encoding Using An Adaptive Delta Modulator With Overshoot Suppression", IEEE Trans. on Commun., Vol. COM-23, Sept. 1975, pp. 905-920.