# RESEARCH ON ADAPTIVE DELTA MODULATORS

Sponsored by the
DEFENSE ADVANCED RESEARCH PROJECTS AGENCY
ARPA Order No. 3534
Contract No. MDA 903-78-C-0182
Monitored by Dr. Vinton Cerf
FINAL REPORT ARPA - FR - 1

Date of Contract 3 Jan. 78
Expiration Date 1 Oct. 79

Period Covered
by this Report 1/3/78-10/1/79



COMMUNICATIONS SYSTEMS LABORATORY
DEPARTMENT OF ELECTRICAL ENGINEERING




THE CITY COLLEGE OF
THE CITY UNIVERSITY of NEW YORK

# DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(14) REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: (18) ARPA - FR-1
2. GOVT ACCESSION NO.: AD-A085 115
3. RECIPIENT'S CATALOG NUMBER:

4. TITLE (and Subtitle): (6) RESEARCH ON ADAPTIVE DELTA MODULATORS
5. TYPE OF REPORT & PERIOD COVERED: Final 3/78 - 10/79
6. PERFORMING ORG. REPORT NUMBER:

7. AUTHOR(s): (10) V. R. Dhadesugoor, C. Ziegler, D. L. Schilling, M. Dressler, S. Davidovici
8. CONTRACT OR GRANT NUMBER(s): (15) MDA-903-78-C-0182, ARPA Order-3534

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
Research Foundation of CUNY on behalf of
The City College
New York, N. Y. 10031
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: (11) 1 Oct 79

11. CONTROLLING OFFICE NAME AND ADDRESS:
ARPA (Code HX 1243)
1400 Wilson Blvd.
Arlington, Va. 22209
(12) 83
12. REPORT DATE: 10/1/79
13. NUMBER OF PAGES: 79

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):
ONR
715 Seventh Avenue
New York, N. Y. 10003
15. SECURITY CLASS. (of this report): Unclassified
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report)
2 copies Director, ARPA, Att: Program Management
1400 Wilson Boulevard
Arlington, Va. 22209
12 copies Defense Documentation Center, Cameron Station, Va. 22314

DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)
(9) Final repts. Mar 78 - Oct 79,

18. SUPPLEMENTARY NOTES
The views and conclusions contained in this document are those of the author and should not be interpreted as necessarily representing the officicial policies, either express or implied, of the Defense Advanced Research Projects Agency or the United States Government

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
(1) Adaptive Delta Modulator
(2) CVSD
(3) Packet Voice
(4) Silence Detection
(5) Slow-scan video
(6) Packet video
(7) Packet loss

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
This report summarizes our results of Voice encoding and video encoding using adaptive delta modulation. Topics include: packet loss, algorithm adaptation, variable rate algorithms, silence detection algorithms, design of a packet voice transmission system, slow-scan video encoding, packet destruction and frame-change detection.

79 12 27 094

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601 |

411 173

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

# RESEARCH ON ADAPTIVE DELTA MODULATORS

Sponsored by the

DEFENSE ADVANCED RESEARCH PROJECTS AGENCY

ARPA Order No. 3534

Contract No. MDA 903-78-C-0182

Monitored by Dr. Vinton Cerf

FINAL REPORT ARPA - FR - 1

Date of Contract 3 Jan. 78

Expiration Date 1 Oct. 79

Period Covered

by this Report 1/3/78 - 10/1/79

Donald L. Schilling

(212) 862-3737

Principal Investigator

COMMUNICATIONS SYSTEMS LABORATORY

DEPARTMENT OF ELECTRICAL ENGINEERING

TABLE OF CONTENTS

# CHAPTER I

## Voice Encoding Using the Adaptive Delta Modulator

### Introduction

In this chapter we discuss the results of experiments that we have performed using adaptive delta modulation as a source encoding technique for use in packet voice networks.

Section 1.1 describes the effect of packet loss as a function of sampling rate, bit error rate and signal level.

Section 1.2 discusses the results obtained when one ADM encoder is used with a different ADM decoder. A simple technique to ensure that the correct encoder/decoder pair is used is described.

Section 1.3 considers the possibility of altering the sampling rate of the ADM encoder upon command and notifying the receiver of this change.

Section 1.4 describes silence-detection algorithms. These algorithms detect the onset of silence and of speech and respectively terminate and start the voice packet.

### Song Voice Adaptive Delta Modulator (SVADM):

The SVADM encoder-decoder (1) is a robust delta modulator system with a dynamic range of 40dB and word intelligibility of 99% at 16Kb/s bit rate and more than 90% of word intelligibility at 9.6Kb/s bit rate. It is easy to implement digitally.

Algorithm:

The algorithm of the SVADM is given by

$$X(k+1) = X(k) + S(k+1) \quad (1)$$

$$S(k+1) = |S(k)|e(k) + S_o e(k-1) \quad (2)$$

and

$$e(k) = \mathrm{Sgn}\left[M(k) - X(k)\right] \tag{3}$$

Where, at $k^{th}$ interval,
$X(k)$ is the estimate of the incoming analog signal,
$S(k)$ is the step size,
$e(k)$ is the digital output of the encoder,
$M(k)$ is the sampled input signal,
$S_o$ is the minimum step size (constant) and

$$\mathrm{Sgn}(y) = \begin{cases} +1 \text{ for } y \geq o \\ -1 \text{ for } y < o \end{cases} \tag{4}$$

Figure 1 shows the block diagram of the SVADM. This algorithm uses a 10 bit arithmetic, i.e. $S(k+1), X(k+1)$ are 10 bit outputs. The minimum step size $S_o \lesssim$ 10mV. The feedback circuit of the encoder is essentially the SVADM decoder. However, in the presence of channel errors, the state of the decoder will be different from that of the encoder. To allow the decoder to attain the state of the encoder, the error correction logic is implemented by modifying the equation (1) for the decoder. The new estimate in the decoder is given by

$$X(k+1) = X(k) + S(k+1) + \beta S_o \tag{5}$$

Let us represent $X(k)$ and $S(k+1)$ by N-bit words so that

$$X(k) = X_o \cdot X_1 X_2 X_3 \cdots X_{N-1} \tag{6}$$

and

$$S(k+1) = S_o \cdot S_1 S_2 S_3 \cdots S_{N-1} \tag{7}$$

Where, $X_o$ and $S_o$ are the sign bits, $X_1$ and $S_1$ are the most significant bits, $X_{N-1}$ and $S_{N-1}$ are the least significant bits of $X(k)$ and $S(k+1)$ respectively.

Then

$$\beta = \begin{cases} +1 \text{ for } X_o = S_o = '1',\ X_1 = X_2 = X_3 = '0' \text{ and } X_{N-1} \oplus S_{N-1} = '0' \\ -1 \text{ for } X_o = S_o = '0',\ X_1 = X_2 = X_3 = '1' \text{ and } X_{N-1} \oplus S_{N-1} = '1' \\ 0 \quad \text{elsewhere} \end{cases} \tag{8}$$

This is known as the leaky Integrator.

**Continuously Variable Slope Delta Modulator (CVSD):**

The CVSD (2) is an adaptive delta modulator specifically designed to process the speech signals. The adaptive technique of the CVSD algorithm exploits the syllabic characteristics of speech so as to minimize the number of bits required in its digital description. Several CVSD processors have been developed. However, the basic principle involving the design of the CVSD is the same. We limit our discussion to outline the principle of operation of the CVSD.

Algorithm:

The general algorithm is given by

$$X(k+1) = aX(k) + \left|(1-a)\ S(k)\right|\ e(k) \tag{9}$$

$$S(k+1) = bS(k) + (1-b)\ (V+V_1) \tag{10}$$

and

$$e(k) = Sgn\left[M(k) - X(k)\right] \tag{11}$$

where, at $k^{th}$ interval,

X(k) is the estimate of the incoming analog signal,

S(k) is the step size,

e(k) is the digital output of the encoder,

M(k) is the input signal,

a is the leak factor associated with the estimate X(k),

b is the leak factor associated with the step size S(k),

V is a constant voltage when three consecutive output bits of the

CVSD encoder are identical (i.e. e(k-2),e(k-1) and e(k) ) and $V_1$ is a constant voltage added to $V_s$ to ensure that the minimum step size is non zero.

Figure 2 shows the block diagram of the CVSD. The values of a and b have been adjusted differently in different CVSD processors. A particular CVSD described in (3) has the values as, a = 0.94 and b = 0.99 at a bit rate, $f_s$, of 16Kb/s.

For our experiments, we have used the CVSD processors developed by the Motorola and the Harris Corporations. We have found subjectively that the quality of the processed speech using the Motorola CVSD is better than that of the Harris CVSD, particularly at input levels of -20db and lower. Therefore, for comparison with the SVADM, we have used the Motorola CVSD.

## The SVADM and the CVSD in the presence of bit errors:

We performed an experiment to compare the SVADM and the CVSD in the presence of bit errors. In order to produce random errors, we used a method shown in Fig. 3. The error generator, shown, consists of a noise generator, a comparator and combinatoric logic. $V_t$ is the threshold voltage of the comparator and is varied to generate different bit error rates. When the noise voltage (Gaussian) exceeds $V_t$, the output of the comparator sets the D flip-flop shown, causing an inversion of the logic state of the transmitted e(k). To determine the error rate, it is necessary to detect the error at every clock cycle and enable a counter to count the total number of errors over a period of time. The error rate is then given by the ratio of the errors counted to the total number of clock periods over the entire counting period.

The CVSD and the SVADM were subjectively compared for bit error rates of $10^{-4}$, $10^{-3}$, $10^{-2}$ and $10^{-1}$. Several listeners were available for the test. With the available comments from them, we were able to establish the results. Figure 4 shows the test set up used for the subjective evaluation. The input speech signal,

from a tape recorder, was bandlimited from 300Hz to 2500Hz, the bit rate was varied from 32Kb/s down to 8Kb/s and the input level was varied from 0dB to -40dB.

The subjective comparison of the CVSD and the SVADM as a function of the sampling rate, $f_s$ and the input level is tabulated in Table 1, when no errors existed. We see from the table that at $f_s$ = 32Kb/s, the speech processed by the SVADM is understandable when the input level is varied from 0dB to -40dB. However, the speech processed by the CVSD loses intelligibility at -40dB input level. Thus, the SVADM offers a 40dB dynamic range, where as, the CVSD offers a 30dB dynamic range at $f_s$ = 32Kb/s. Similarly we also see from the table that at $f_s$ = 16Kb/s, the SVADM offers a 30dB dynamic range and the CVSD offers a 20dB dynamic range. At $f_s$ = 9.6Kb/s, the SVADM has a 20dB dynamic range and the CVSD has a 10dB dynamic range. Figure 5 shows the dynamic ranges of the SVADM and the CVSD as a function of $f_s$ when no errors existed.

In the presence of bit errors, the dynamic ranges of both the SVADM and the CVSD varied as a function of bit error rate. Figure 6 shows the dynamic ranges of the SVADM and the CVSD as a function of bit error rates. We see from Fig. 6, the SVADM offers a 10dB higher dynamic range over the CVSD at different error rates. This is true at different bit rates. Table 2 shows the subjective comparison of the CVSD and the SVADM at different error rates and bit rates. We see from the table that the CVSD was preferred to the SVADM at $f_s$ = 32Kb/s and the error rate of $10^{-1}$ for a 0dB input level. Under all other conditions of operation, the SVADM was preferred to the CVSD. The SVADM is significantly better than the CVSD, particularly at input levels of -20dB and lower. This is true for all bit rates and bit error rates.

## DELTA MODULATORS IN A PACKET VOICE NETWORK

Current methods used for digitizing voice in packet voice networks are the Pulse Code Modulation (PCM), Adaptive Delta Modulation (ADM) and the Linear Predictive Coding (LPC). If PCM

is used to encode 2.5KHz voice, one would require a bit rate of atleast 40Kb/s to produce good quality voice. A packet size of 1000 bits requires that the PCM packets be transmitted at the rate of 40 Packets/sec. The ADM systems reproduce good quality voice, when operated at 10-16Kb/s. For the same packet size, the ADM packets can be transmitted at the rate of 10-16 packets/sec. The ADM is also preferred to the LPC, since the LPC is still a relatively high cost and complex system. The ARPA network is currently employing the CVSD algorithm to digitize voice. Therefore, it is appropriate to compare the use of the SVADM with that of the CVSD in a packet voice network. We have already shown that the performance of the SVADM is preferred to that of the CVSD when operated at bit rates of 16Kb/s and lower. We have compared the performance of the SVADM in a packet voice network, interms of packet size (P), bit rate ($f_s$) and packet loss rate (r), with that of the CVSD.

## 1.1 Packet Loss

### Concept of Packet Loss:

In a packet switched network, when a customer A (source) asks for a connection to a called party B (destination), the customer's packets are then transmitted, interleaved with other packets from one exchange to another, thus giving a "Virtual" connection". Once the contact has been established between the source A and the destination B, B would be receiving a virtually continuous stream of packets as long as A is active. As the packets arrive, the destination B processes them. Thus while the $i^{th}$ packet is being processed, the destination B looks for $(i+1)^{st}$ packet. If the $(i+1)^{st}$ packet is not available for processing after B has completed processing the $i^{th}$ packet, then we recognize the $(i+1)^{st}$ packet as being lost. In a normal operation, the destination B can lose the $(i+1)^{st}$ packet in one of two different ways as follows:

(a) The $(i+1)^{st}$ packet actually arrived at B, but was rejected as non valid. When a non valid packet is received, the request for

retransmission is not required in voice transmission since voice systems using delta modulators generally tolerate reasonable error rates and besides, the delay constraints preclude the use of retransmission of packets anyway.

(b) The $(i+1)^{st}$ packet has not arrived (i.e. it is late) even after B has completed processing the $i^{th}$ packet. After waiting for an appropriate period, the destination B, then, will decide that the $(i+1)^{st}$ packet is lost and starts looking for the $(i+2)^{nd}$ packet.

Effect of Packet Loss:

When the destination B decides that a packet is lost and starts processing the next packet, the reproduced speech signal exibits a loss of speech. If, for example, the speech is encoded at 16Kb/s and the packet size is 1Kbits, the fraction of the speech lost due to a single packet loss is $(1/16)^{th}$ of a second or approximately 60msec. The degradation of the quality of the speech processed due to 60msec. of speech loss, is minimal. This is because the human ear is insensitive to the small amount of degradation. Also, if one of every hundred packets is lost, then 60msec. of speech loss occurs in 6 seconds of speech and this too does not adversely affect the quality of the processed speech.

When a packet is lost, the state of the delta modulator decoder (similar to bit error described earlier) is different from that of the encoder. However, this will be corrected by the error correction logic as described earlier (refer to Equations (4)-(7)).

COMPENSATION ALGORITHMS AT THE RECEIVER - -

In addition to the earlier mentioned error correction technique, in order to help the receiver in its correction process, we have developed compensation algorithms for use by the receiver during the length of the packet loss. Three different compensation algorithms have been studied.

Algorithm 1: Freeze the decoder.

In this algorithm, the state of the receiver remains constant or is frozen during the packet loss period. This is done by inhibiting the sampling clock pulse to the decoder during the entire length of the missing packet. This enables the decoder to remain at the same state; that is the receiver step size and estimate remain the same until a new packet is received. The encoder, however, is changing its state continuously. Thus, the state of the decoder is different from that of the encoder when the new packet arrives. This will be eventually corrected by the leaky integrator error correction routine. During a packet loss, freezing the receiver usually creates a large step size error.

Algorithm 2: Generate a local periodic 11001100··· steady state pattern at the receiver.

In this method, the receiver will locally generate a 11001100··· pattern for the entire packet loss period. The steady state pattern at the decoder input, would enable the receiver estimate to leak to zero level, during the period of a lost packet. However, the step size error remains unchanged. It must be noted that the steady state pattern 11001100··· is only applicable to the SVADM decoder and not the CVSD decoder. This steady state pattern of 11001100··· generates an oscillation at $f_s/4$ and usually is heard at low bit rates.

Algorithm 3: Generate a local periodic 101010··· steady state pattern at the receiver.

In this algorithm, the receiver will locally generate a 101010··· pattern instead of 11001100··· as in algorithm 2. This pattern at the input of the decoder enables the stepsize to become smaller. However, the estimate error remains approximately the same. The smaller step size in the decoder is extremely

advantageous, since it will prevent any large variation of the magnitude of speech due to an error at the input. This is particularly more pronounced at high error rates. In addition, at low bit rates, the oscillations at $f_s/2$ is not heard. Even though, the step size due to this decoder reaches a minimum, the adaptive step size algorithm enables the decoder step size to grow fast once the new packets are processed.

Figure 7 displays the receiver estimates obtained during a packet loss period, using the three methods.

## EXPERIMENT FOR PACKET LOSS STUDIES

The test set up used for packet loss studies is illustrated in Fig. 8. The input speech was bandlimited from 300Hz to 2500Hz. The packet errors are generated by using the method shown in Fig.3 except that we checked for an error only once in a given packet. When a random error occurs, the entire packet is not transmitted. The input speech signal was encoded by the SVADM and the CVSD encoders. The two encoders' output bits were then packetized. The packetizers, packet loss generation and the depacketizers were simulated using a PDP-11/34 computer for real time operation. The outputs of the depacketizers were then decoded respectively by the SVADM and the CVSD decoders and the processed speech signals were bandlimited from 300Hz to 2500Hz and heard by using head sets. Two types of speech tapes were used.

1. A Mark Twain story
2. A general radio conversation.

All three receiver compensation algorithms for packet loss were tested using the SVADM encoder-decoder and algorithm 1 and 3 were tested using the CVSD encoder-decoder, since the steady state output pattern for the CVSD is 101010···. The parameters for the subjective quality test are the packet size P, the packet loss rate r and the bit rate $f_s$.

## RESULTS

The subjective comparison of the CVSD and the SVADM in terms of P,r and $f_s$ for OdB input level is tabulated in Table 3. At the maximum input level (OdB), the performance of the packet voice system using the SVADM or the CVSD was found to be about the same. However, at lower input levels, there is a general degradation in the performance of the CVSD as found to be true earlier (refer to Table 1 and 2).

There was no difference in the performance regarding the intelligibility using the three receiver algorithms for packet loss. However, for the SVADM encoder-decoder, using the receiver compensation algorithms 1 and 2, when a packet loss occured, there was a large change in the estimated speech due to large step size errors. This change in the estimate sometimes was annoying to the listeners, particularly at high packet loss rates ($r = 10^{-1}$). This effect, however, was not found when using the receiver compensation algorithm 3.

As seen from the Table 3, a loss rate of $10^{-2}$ was not noticeable. The breaks in speech were distinguishable only at loss rates of $10^{-1}$ and $2(10^{-1})$. However, the speech was intelligible even at loss rates of $10^{-1}$. This result is true for packet sizes of P = 2048, 1024, 512, 256 and $f_s$ = 16 and 9.6 KB/s.

## CONCLUSIONS

From our experiments, we derived the following conclusions:

(a) A packet loss rate up to $10^{-2}$ is not noticeable.

(b) At packet sizes of 2048, 1024 bits and $f_s$ = 16Kb/s, the talk spurt break of 128msec. and 64msec. respectively for a single packet loss is noticed predominantly at loss rates of $10^{-1}$ and $2(10^{-1})$. This is true because of the fact that the human ear notices any loss of speech over 30msec. duration. However, the overall intelligibility was still acceptable.

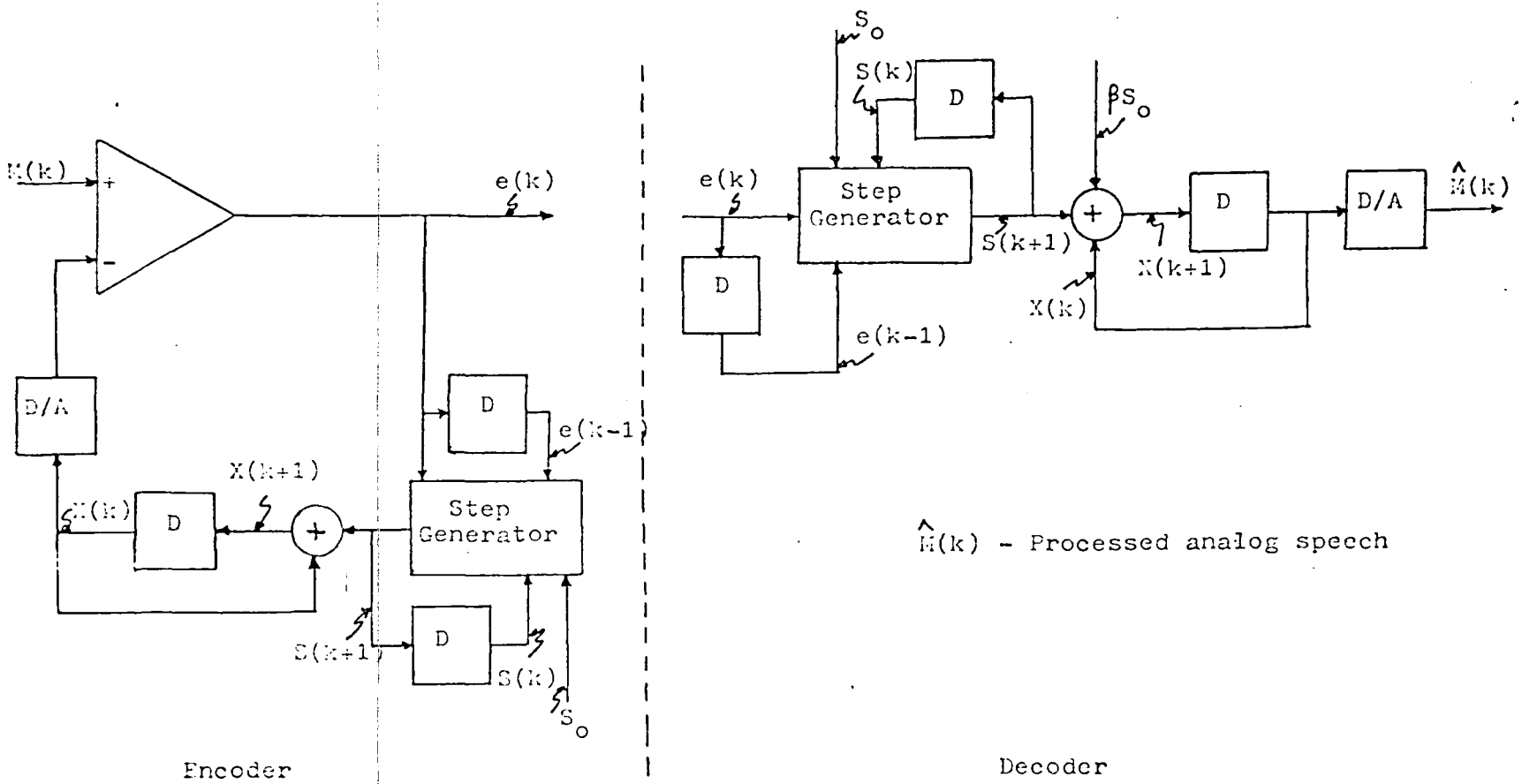
(c) The results show that packet switching network using delta modulation source encoders can safely operate at loss rates of $10^{-2}$

REFERENCES:

1) C.L. Song, J. Garodnick, D.L.Schilling, "A variable step size robust delta modulator ", IEEE, COM-TECH, Vol.19, pp 1033-1040, Dec. 1971.

2) R.E.Crochiere, D.J.Goodman, L.R.Rabiner, M.R.Sambur, "Tandem connections of wide band and narrow band speech communication systems, Part I - Narrow band to wide band link", B.S.T.J., Vol. 56, No.9, Nov. 1977.

3) Motorola Incorporated, "Semiconductor data library", Vol. 6/ series B, pp 130-134.




$\hat{M}(k)$ - Processed analog speech

Fig. 1 Block diagram of the SVADM

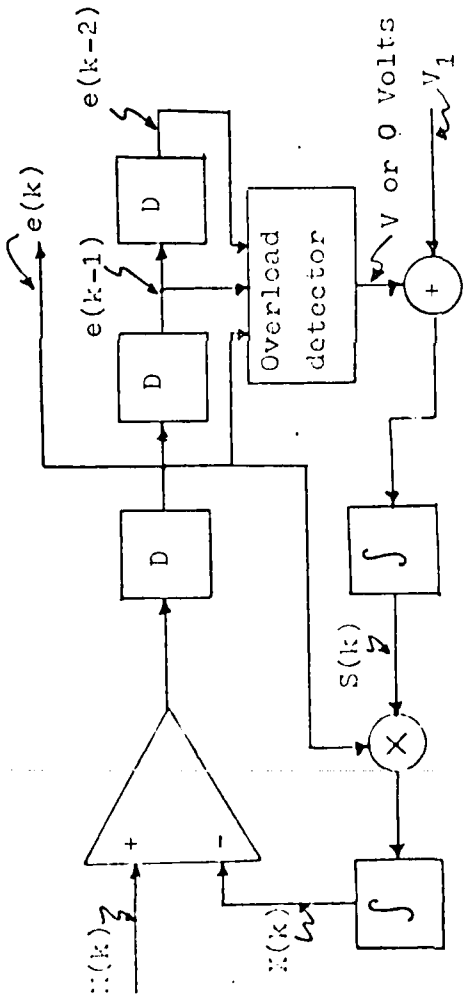



Fig. 2 Block diagram of the CVSD

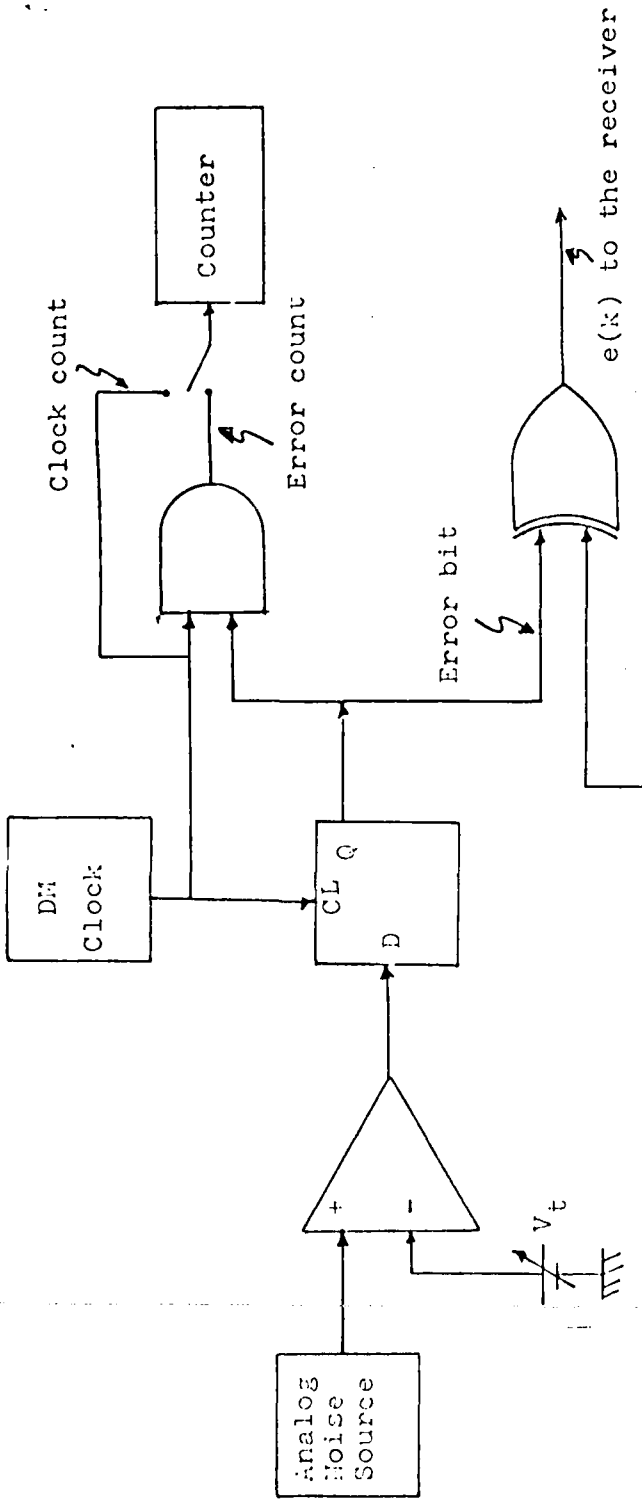



Fig. 3 Generation of random errors

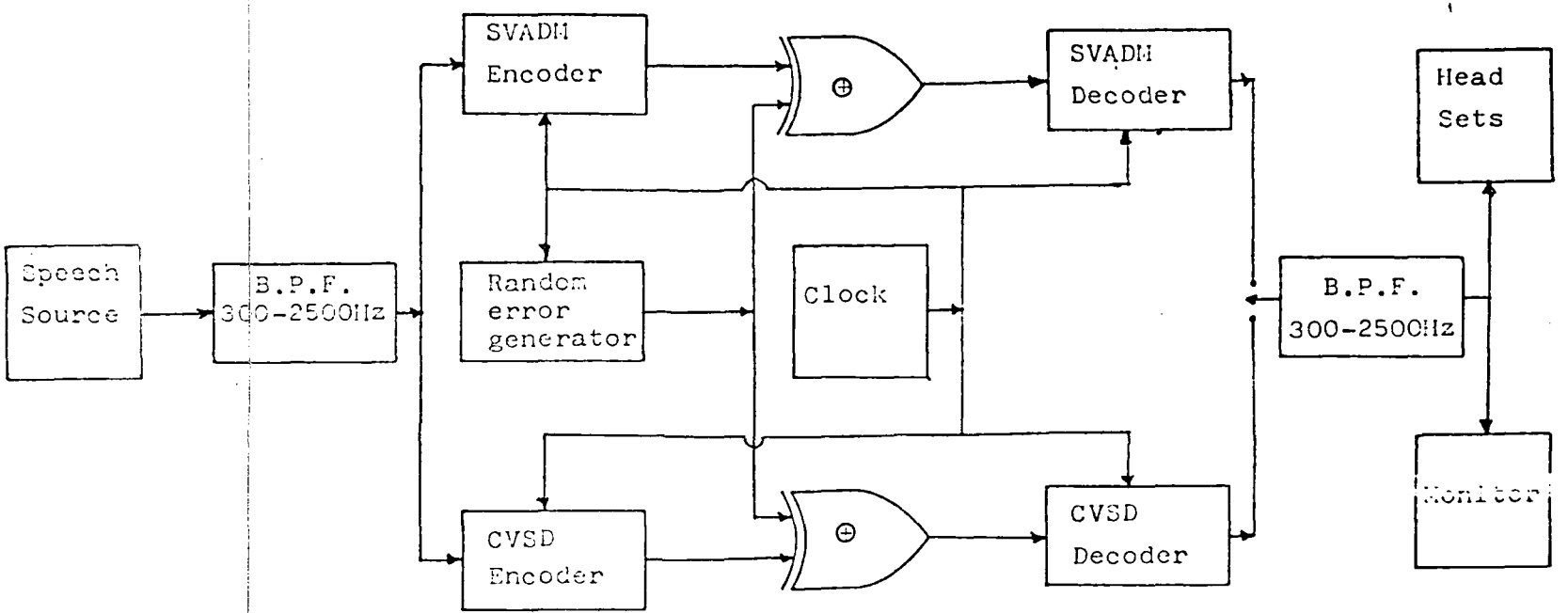



Fig. 4 Test set up for comparison of the CVSD and the SVADM in the presence of bit errors.

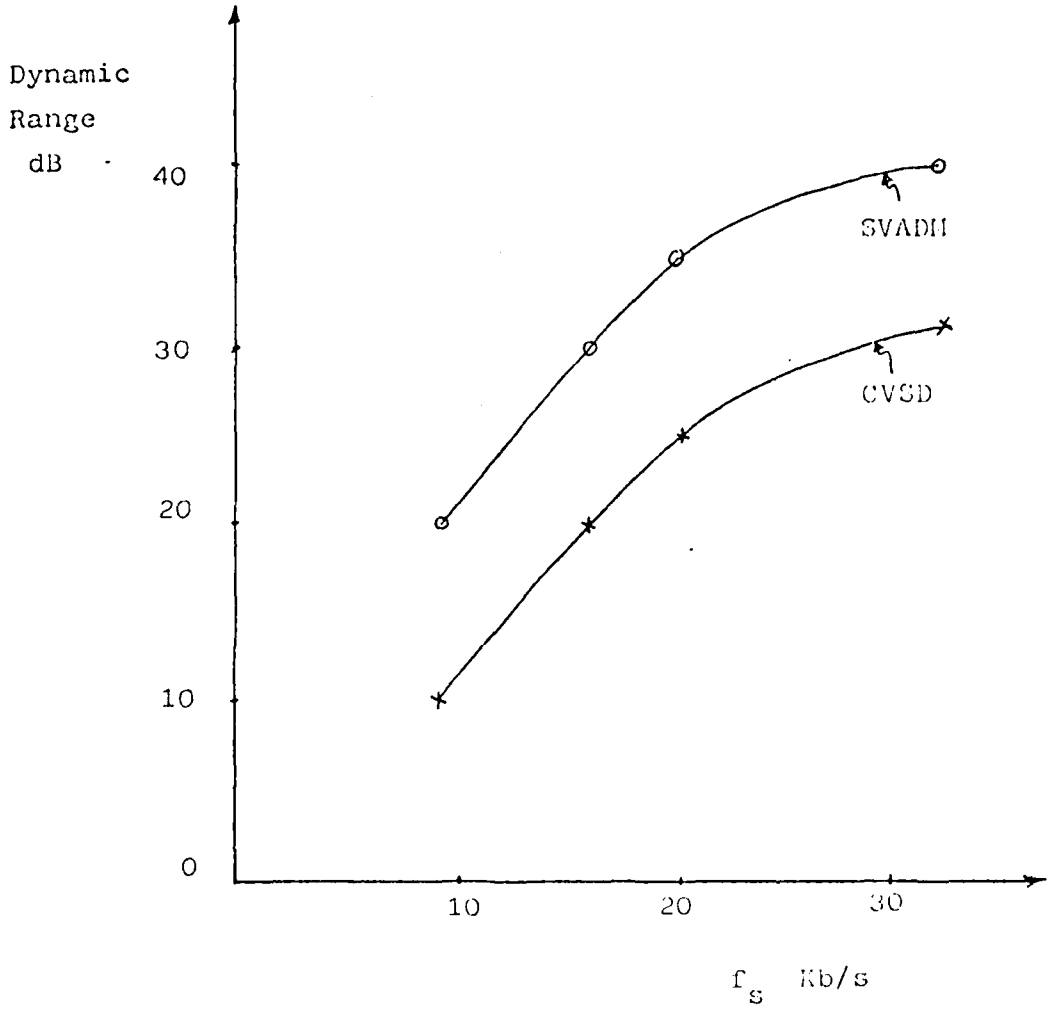
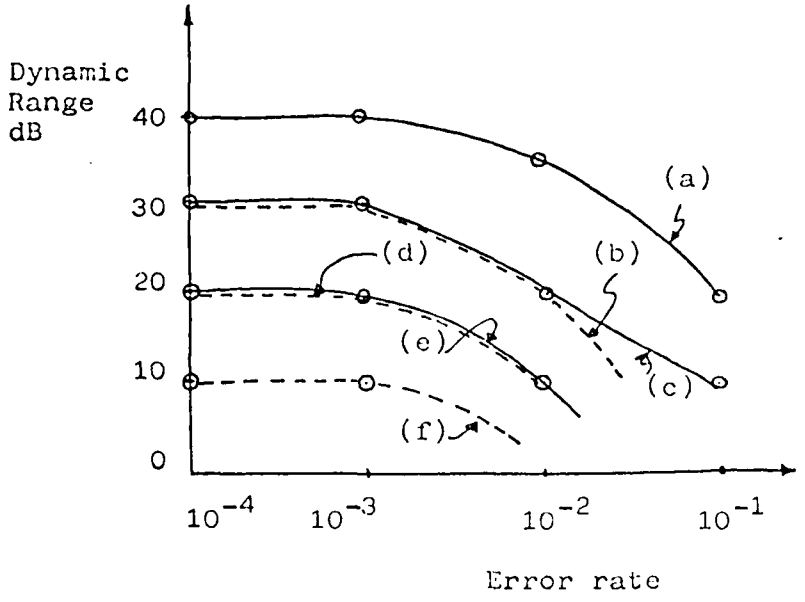



Fig. 5 Dynamic range as a function of bit rate




(a) - SVADM at $f_s$= 32 Kb/s

(b) - CVSD at $f_s$= 32 Kb/s

(c) - SVADM at $f_s$= 16 Kb/s

(d) - CVSD at $f_s$= 16 Kb/s

(e) - SVADM at $f_s$= 9.6 Kb/s

(f) - CVSD at $f_s$= 9.6 Kb/s

Fig. 6 Dynamic range as a function of error rate.

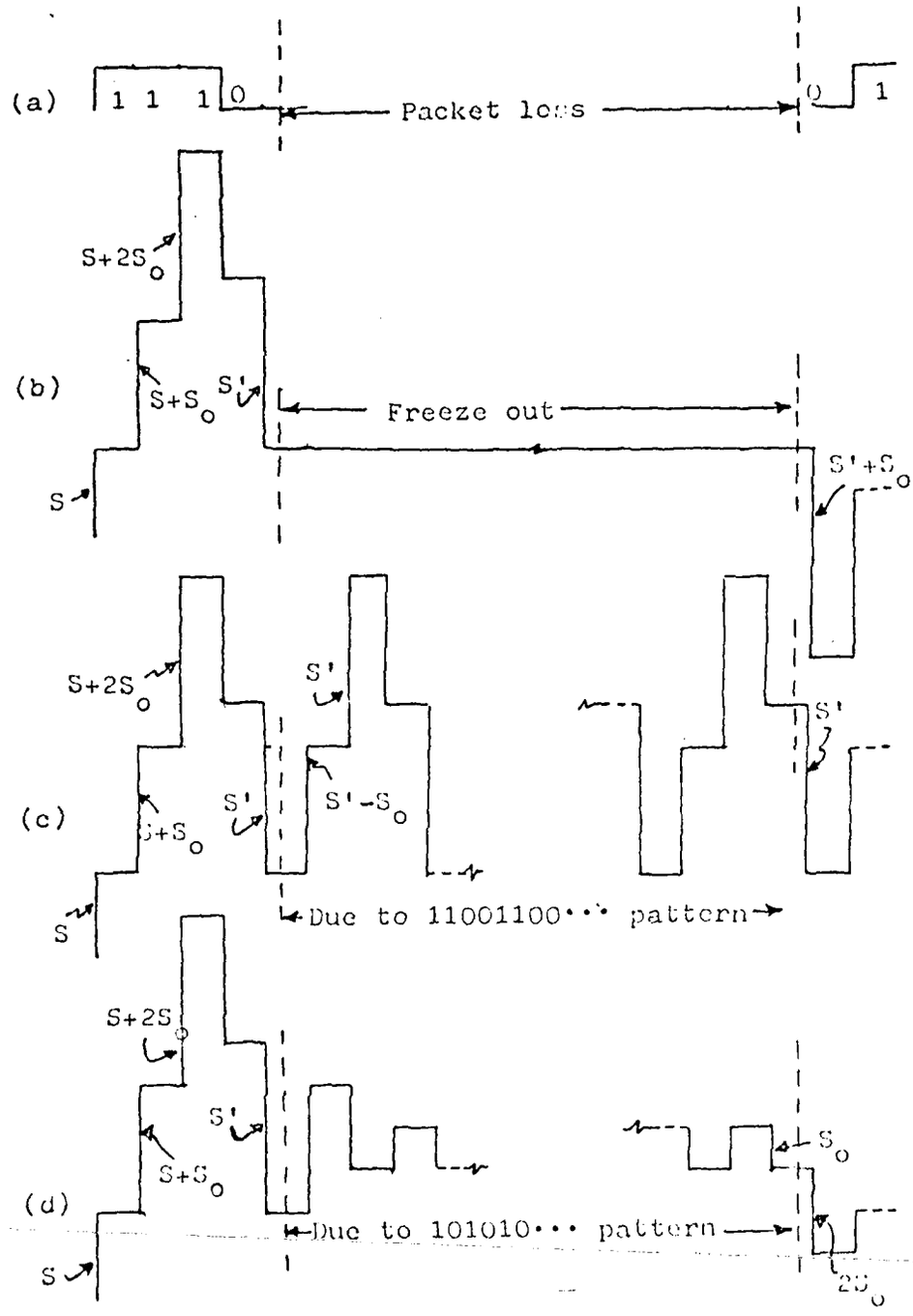
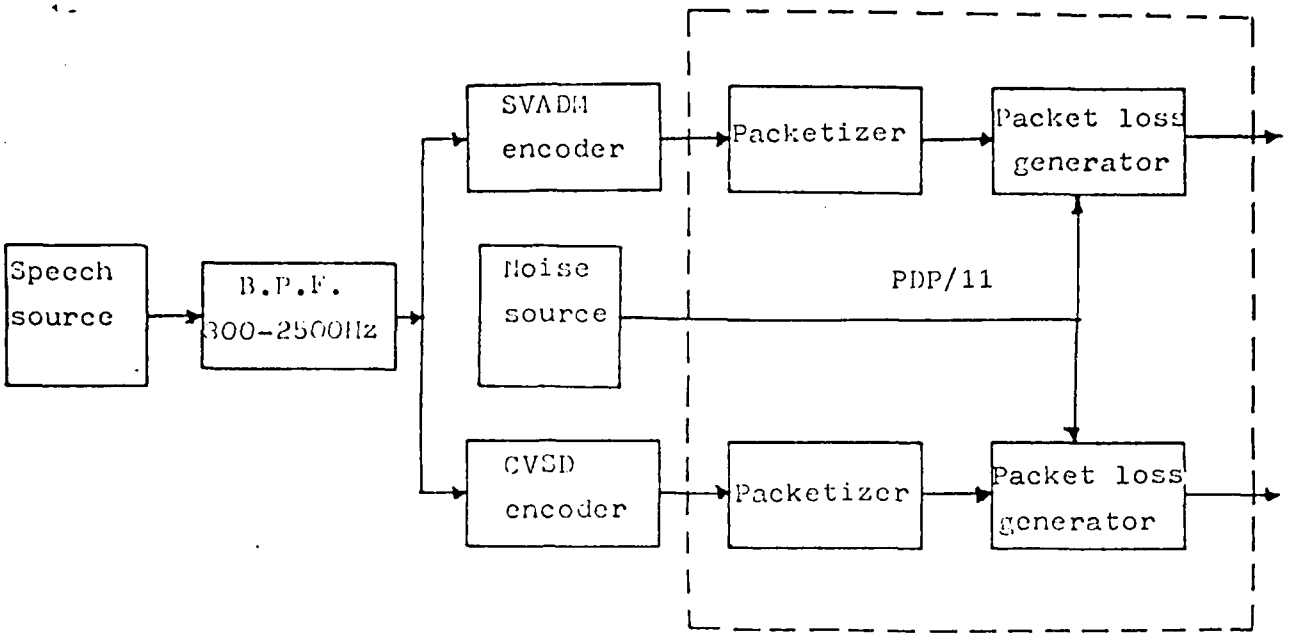
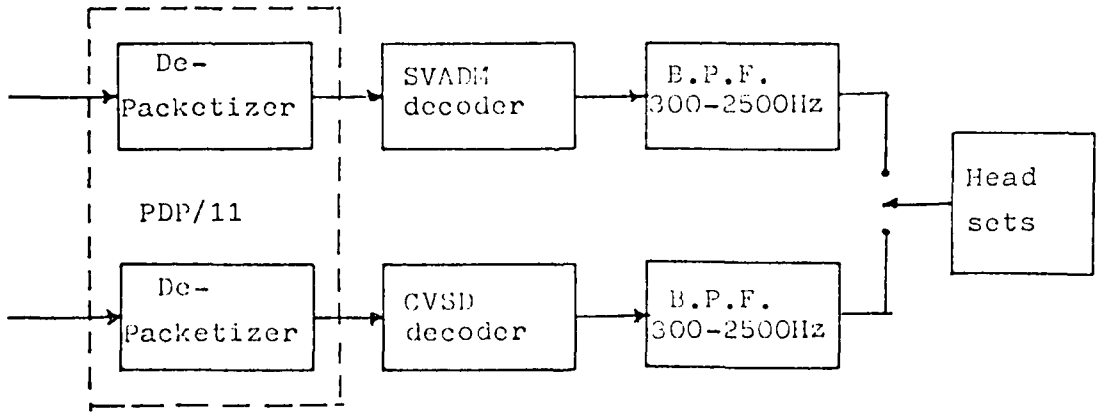



Fig. 7 Estimates using algorithm 1, 2 and 3.

(a) Input bit pattern at the decoder,
(b) Estimate using algorithm 1,
(c) Estimate using algorithm 2, and
(d) Estimate using algorithm 3.




(a)




(b)

Fig. 8 Test set up for packet loss studies.
(a) Transmitter section,
(b) Receiver section.

Table 1: Subjective comparison of dynamic ranges of CVSD and SVADM.

| $f_s$ Kb/s | Input dB | CVSD | SVADM | Comparison |
|---|---|---|---|---|
| 32 | 0 | Intelligible | Intelligible | No difference |
| | -10 | Intelligible | Intelligible | No difference |
| | -20 | Intelligible | Intelligible | No difference |
| | -30 | Intelligible: Voice is breaking and buzzy | Intelligible | SVADM preferred |
| | -40 | Not intelligible | Intelligible: voice is buzzy | SVADM preferred |
| 16 | 0 | Intelligible | Intelligible | No difference |
| | -10 | Intelligible | Intelligible | SVADM preferred |
| | -20 | Intelligible: Voice is buzzy | Intelligible | SVADM preferred |
| | -30 | Not intelligible | Intelligible: has back-ground noise | SVADM preferred |
| | -40 | Not intelligible | Not intelligible | |
| 9.6 | 0 | Intelligible | Intelligible | No difference Granularity due to $f_s$ exists |
| | -10 | Not intelligible | Intelligible | SVADM preferred |
| | -20 | Not intelligible | Intelligible: Noisy | SVADM preferred |
| | -30,-40 | Not intelligible | Not intel-ligible | |

Table 2: Subjective comparison of the CVSD and the SVADM at different error rates.

| $f_s$ Kb/s | Input level dB | Error rate | CVSD | SVADM | Comparison |
|---|---|---|---|---|---|
| 32 | 0 | $10^{-4}$ | Intelligible: Same as at no errors | Intelligible: Same as at no errors | No preference |
| | | $10^{-3}$ | Intelligible: Same as at no errors | Intelligible: Same as at no errors | No preference |
| | | $10^{-2}$ | Intelligible: With back ground noise (smearing) | Intelligible: With back ground noise | No preference |
| | | $10^{-1}$ | Intelligible: More noise | Intelligible: More noise | CVSD preferred |
| 32 | -20 | $10^{-4}$ | Intelligible: Same as at no errors | Intelligible: Same as at no errors | No preference |
| | | $10^{-3}$ | Barely intelligible | Intelligible: Same as at no errors | SVADM preferred |
| | | $10^{-2}$ | Barely intelligible | Intelligible | SVADM preferred |
| | | $10^{-1}$ | Not intelligible | Not intelligible | |
| 16 | 0 | $10^{-4}$ | Intelligible: Same as at no errors | Intelligible: Same as at no errors | No preference |
| | | $10^{-3}$ | Intelligible: Same as at no errors | Intelligible: Same as at no errors | No preference |

contd.

Table 2: continued

| $f_s$ Kb/s | Input level dB | Error rate | CVSD | SVADM Comparison |
|---|---|---|---|---|
| 16 | 0 | $10^{-2}$ | Intelligible | Intelligible SVADM preferred |
| | | $10^{-1}$ | Not intelligible | Not intelligible |
| 16 | -20 | $10^{-4}$ | Intelligible: Same as at no errors | Intelligible: No preference Same as at no errors |
| | | $10^{-3}$ | Barely intelligible | Intelligible SVADM preferred |
| | | $10^{-2}$ | Barely intelligible: Not Acceptable. Heavy background noise | Intelligible: SVADM preferred Fluttering noise |
| | | $10^{-1}$ | Not intelligible | Not intelligible |
| 9.6 | 0 | $10^{-4}$ | Intelligible: Same as at no errors | Intelligible: No preference Same as at no errors |
| | | $10^{-3}$ | Intelligible: Same as at no errors | Intelligible: No preference Same as at no errors |
| | | $10^{-2}$ | Intelligible: Noisy | Intelligible: No preference Noisy |
| | | $10^{-1}$ | Not intelligible | Not intelligible |
| 9.6 | -20 | $10^{-4}$ | Intelligible: Same as at no errors | Intelligible: No preference Same as at no errors |

Table 2: continued

| $f_s$ Kb/s | Input level dB | Error rate | CVSD | SVADM | Comparison |
|---|---|---|---|---|---|
| 9.6 | -20 | $10^{-3}$ | Not intelligible | Intelligible | SVADM preferred |
| | | $10^{-2}$, $10^{-1}$ | Not intelligible | Not intelligible | |

Table 3: Subjective comparison of the CVSD and the SVADM in terms of P, $f_s$, and r.

Criteria for comparison:

1. Intelligibility - The speech is intelligible if it is understandable.
2. Acceptability - The speech is acceptable if words or syllables are not missing.

| P bits | $f_s$ Kb/s | r | CVSD | SVADM | Comparison |
|---|---|---|---|---|---|
| 2048 | 16 | 0 | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. |
| | | $10^{-4}$, $10^{-3}$, $10^{-2}$. | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Performances are similar to when r = 0. |
| | | $10^{-1}$ | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Breaks in speech are noticed. |
| | | $2x(10^{-1})$ | Intelligible. Not acceptable. | Intelligible. Not acceptable. | - |
| 2048 | 9.6 | 0 | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. |
| | | $10^{-4}$, $10^{-3}$, $10^{-2}$. | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Performances are similar to when r = 0. |
| | | $10^{-1}$ | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Breaks in speech are noticed. |
| | | $2x(10^{-1})$ | Not intelligible Not acceptable. | Not intelligible Not acceptable. | - |

Table 3: continued

| P bits | $f_s$ Kb/s | r | CVSD | SVADM | Comparison |
|---|---|---|---|---|---|
| 1024 | 16 | 0 | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. |
| | | $10^{-4}, 10^{-3}, 10^{-2}$. | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Performances are similar to when r = 0. |
| | | $10^{-1}$ | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Breaks in speech are noticed. |
| | | $2x(10^{-1})$ | Intelligible. Not acceptable. | Intelligible. Not acceptable. | - |
| 1024 | 9.6 | 0 | Intelligible. Acceptable. | Intelligible. Acceptable. | No preferncc. Granularity is heard. |
| | | $10^{-4}, 10^{-3}, 10^{-2}$. | Intelligible Acceptable. | Intelligible. Acceptable. | No preference. Performances are similar to when r = 0. |
| | | $10^{-1}$ | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Breaks in speech are noticed. |
| | | $2x(10^{-1})$ | Not intelligible. Not acceptable. | Not intelligible. Not acceptable. | - |
| 512 | 16 | 0 | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. |
| | | $10^{-4}, 10^{-3}, 10^{-2}$. | Intelligible. Acceptable. | Intelligible. Acceptable. | No preferncc. Performances are similar to when r = 0. |

Table 3: continued

| P bits | $f_s$ Kb/s | r | CVSD | SVADM | Comparison |
|---|---|---|---|---|---|
| 512 | 16 | $10^{-1}$ | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Breaks in speech are noticed. |
| | | $2x(10^{-1})$ | Intelligible. Not acceptable. | Intelligible. Not acceptable. | - |
| 512 | 9.6 | 0 | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Granularity is heard. |
| | | $10^{-4}, 10^{-3}, 10^{-2}$. | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Performances are similar to when r = 0. |
| | | $10^{-1}$ | Intelligible.. Acceptable. | Intelligible. Acceptable. | No preference. Breaks in speech are noticed. |
| | | $2x(10^{-1})$ | Not intelligible. Not acceptable. | Not intelligible. Not acceptable. | - |
| 256 | 16 | 0 | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. |
| | | $10^{-4}, 10^{-3}, 10^{-2}$. | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Performances are similar to when r = 0. |
| | | $10^{-1}$ | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Breaks in speech are noticed. |

Table 3: continued

| P bits | $f_s$ Kb/s | r | CVSD | SVADM | Comparison |
|---|---|---|---|---|---|
| 256 | 16 | $2x(10^{-1})$ | Intelligible. Not acceptable. | Intelligible. Not accepatble. | - |
| 256 | 9.6 | 0 | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Granularity is heard. |
| | | $10^{-4}, 10^{-3}, 10^{-2}$. | Intelligible. Acceptable. | Intelligible. Acceptable. | No preferences. Performance are similar to when r = 0. |
| | | $10^{-1}$ | Intelligible. Acceptable. | Intelligible. Acceptable. | No preference. Breaks in speech are noticed. |
| | | $2x(10^{-1})$ | Not intelligible. Not acceptable. | Not intelligible. Not acceptable. | - |

## 1.2 Algorithm Adaptation

Every Company, Country, even U.S. Government Agency has its own favorite adaptive deltamodulator. In most cases it is possible to insure that transmitters and receivers each use the same ADM system, however, on occasion the communication net may be so vast that an ADM encoder can be used in a transmitter and a different ADM decoder might be present in the decoder. A similar problem continually arises in PCM where the U.S. and Canada use a u-Law Companding technique while Europe and the rest of the world use the A-Law Companding technique. To communicate between the U.S. and Europe requires an interface to couple the systems or as is more common, the transmitter uses the receiver's algorithm. However, it has been found that in a single link there is no increase in error rate if a u-Law encoder is used with an A-Law decoder or vice-versa. This very interesting result derives from the similarity between algorithms. Similarly most ADM's "look" alike.

An experiment was performed in which the SONG ADM or CVSD ADM was used as an encoder and another model ADM or an RC low pass filter used as the decoder. The RC filter showed the greater degradation in each case, however, at 32kb/s and at 16 kb/s the voice was completely intelligible and completely recognizeable.

In many cases such degradation is intolerable. In these conditions it is possible, in the packet protocol, to specify the algorithm. This is readily done, for the CVSD transmits a steady state pattern of ...1010... while the SONG ADM has a steady state pattern ...1100.... However any other code is adequate. A correlator in the receiver recognizes the code and connects the appropriate decoder into the circuit.

It is interesting to note that even when the correct decoder is employed the signal suffers degradation. This

phenomenon does not occur using PCM. Experiments performed indicate that no more than 3 A/D - D/A conversions can be cascaded when using the CVSD at 16kb/s. The SONG ADM can sustain four such conversions.

## 1.3 Variable-Rate Algorithms

The quality of voice obtained from an ADM operating at 32 kb/s is far superior to the quality at 16 kb/s which, in turn, is superior to the quality at say 8 kb/s. Below 8 kb/s the ADM quality degrades extremely rapidly.

When a communication channel is being lightly used it would be nice if we could transmit the ADM encoded voice at 32 kb/s providing that when the channel becomes congested we could sample at say 8 kb/s. For example, in a practical system we might be required to pass high priority data lasting for say, 1 second. The degradation of the voice quality during this interval, caused by dropping the sampling rate would not even be noticed if the bursts of data were spaced relatively far apart. As a matter of fact we saw that completely losing 1 packet in 10 was needed before the packet loss became noticeable. Here, we are not losing packets but degrading performance.

In any variable rate radio system some bit synchronization must be present in the receiver to lock to the transmitted clock. If two or three frequencies such as 8,16 and 32 kb/s are employed the bit synchronizer is constructed from a single clock and stability is assured.

## 1.4 Silence Detection Algorithms

### Introduction

Past research has shown that roughly 50% of conversational speech consists of silent periods; that is, time in

which no speaker is actually talking. Hence, in order to reduce the total packet transmission rate in packet voice systems, it would clearly be advantageous to detect these silent periods and not transmit any packets during these times.

Using delta modulation techniques, such as the Song Voice Adaptive Delta Modulator (SVADM) or the Continuously Variable Slope Delta Modulator (CVSD) we have devised and experimentally tested algorithms for digital detection of silent periods. The algorithms are based on the fact that during silent or steady-state periods, these delta modulators will exhibit a periodic pattern. Using this knowledge, one can then analyze the bits in a given voice packet and determine how much of the packet was silent. Then upon setting a threshold, one decides whether a given packet contains enough information to be transmitted or whether the packet is from a silent period and should not be transmitted.

Real time experiments were performed to test the quality of speech obtained while employing the silence detection algorithm. The parameters of the experiments were sampling rate packet size and threshold level. In addition, algorithms for use by the receiver during these silent periods, periods in which it receives no packets, were developed. Three different algorithms were tried and compared. Finally, the notion of repacking was developed. By repacking, we refer to the idea where the transmitter, having detected that it is currently in a silent period, will halt its packetization process until such time as it detects the initiation of a new speech period. Only then will the transmitter begin the formation of a new packet. It was found that repacking vastly enhances the quality of the received speech.

**From the results of our experiments, we have concluded that the digital silence detection techniques we have developed may be used at threshold levels so as to eliminate nearly all the silent packets from transmission without loss of any signifcant quality to the received speech.**

## ALGORITHM FOR SILENCE DETECTION

The SVADM produces a 11001100··· pattern in the steady state for a constant input. On the other hand, the CVSD produces a 10101010··· pattern. In order to detect the onset of silence, we shall employ an algorithm which will detect these steady state patterns.

To determine the start of a silent period for the SVADM, we observe eight consecutive bits of the encoder output to see if they have a 11001100 pattern (or any of the three other possible permutations of a 11001100 for eight bits as illustrated in Fig.3). If this pattern is detected, a decision that a silent period has begun is made. The reason for choosing eight bits for detection of silence rather than four consecutive bits is due to the fact that the SVADM encoder output might have a 1100 or any one of the other permutations at the peak of the input signal and thus create false silence periods. Also, we have found that when the input signal varies over the full dynamic range, no difference exists, whether we use eight or twelve consecutive bits for detection of silence. Thus, we have used a minimum of eight consecutive bits to detect the onset of silence.

Having entered a silent period, the silent period will be said to end when three consecutive encoder outputs are identical (i.e. either 000 or 111). The SVADM produces a minimum of three bits of 000 or 111 at the onset of speech. It is obvious that the detection of the onset of speech is not feasible using only two bits of the same sign due to the form of the steady state pattern. Also, using more than three consecutive bits of of the same sign may cause the

initial part of the speech to be clipped.

For detecting the onset of silence in the case of CVSD encoder, we look for eight bits of 10101010, since the output of the CVSD encoder in the steady state is 101010···. Here too, we remain in the silent period until the three consecutive bits of 111 or 000 are detected for speech initiation. Figure 4 shows the timing diagram for silence detection and speech initiation.

## 1. Silent packets

As the transmitter assembles a packet, we keep track of the number of silence (steady state) bits by using a counter. To determine whether a packet is silent or not, we set up a threshold parameter $T_p$, which is a number assigned to a packet. If the ratio of the number of silence bits, S, to the total number of bits in a packet, P, exceeds $T_p$, then we say, the packet is a silent packet, i.e., we consider this packet not to have enough useful information to make it worthy of transmission. As such, all silent packets are not transmitted. Clearly, this reduces the packet transmission rate. Figure 5 shows the discarding of silent packets. In Fig. 5, $p_1$ and $p_5$ are speech packets, $p_2$ and $p_4$ are silent packets since $S/P \geq T_p$ and $S''/P \geq T_p$ respectively. However, $S'/P < T_p$ and therefore, the packet $p_3$ is not a silent packet. In this case only $p_1$, $p_3$ and $p_5$ are transmitted.

By not transmitting $p_2$ and $p_4$, we loose some speech bits. For example, the initial part of the speech in $p_2$ is lost. Experiments have been performed to evaluate the effect of such a loss of speech during transmission. The result will be presented later.

We have described the process of packetization and determination of silent packets. The packet size is kept constant and packetization is performed for every P consecutive bits. We refer to this method of packetization as Non-Repacking. Another scheme we have used is called Repacking.

2. Repacking:

By repacking, we refer to the idea in which the transmitter, having currently detected a silent period, halts its packetization process until such time as it detects the initiation of a new speech period. Only then, will the transmitter begin the formation of a new packet. Figure 6(a) and (b) illustrate the non-repacking and the repacking schemes respectively.

In Fig. 6(a), we show that $p_1$, $p_2$, $p_3$, $p_4$ and $p_5$ are packets of size P bits. The shaded area corresponding to S, S', S" represent silence bits in each of the packets $p_2$, $p_3$ and $p_4$ respectively. $p_1$ and $p_5$ are speech packets. $p_2$, $p_3$ and $p_4$ are silent packets, since $(S/P) \geq T_p$, $(S'/P) \geq T_p$ and $(S''/P) \geq T_p$. Thus only $p_1$ and $p_5$ are transmitted. By not transmitting $p_2$, $p_3$ and $p_4$, some speech bits are lost in those packets. The speech bits lost in $p_4$ can be saved if the repacking scheme is used as shown in Fig. 6(b).

In the repacking scheme, after determining that $p_2$ and $p_3$ are silent packets, the transmitter recognizes that the encoder output still has silent bits and therefore will halt its packetization process. It will start packetization once it detects that the speech has been initiated and therefore the new packet is now $p_4'$ and not $p_4$. Thus the repacking scheme transmits the speech bits contained in $p_4$ which was lost when the non-repacking scheme was used. Therefore, in the repacking scheme, there is less chance of losing the onset of speech. However, the speech bits lost in $p_2$ and $p_3$ cannot be recovered in either of the schemes. It was found that repacking vastly enhances the quality of the processed speech.

## COMPENSATION ALGORITHMS DURING SILENT PERIODS

When the transmitter decides that a packet (silent packet) is not worthy of transmission, it will not send the packet. When the silent periods are not transmitted, a gap is created in the stream of received packets at the receiver. At this point, the receiver recognizes that a silent period has begun at the source. As such, it will

now begin to take local compensating action. Three different compensating algorithms have been studied.

1. Algorithm 1: Freeze the decoder.

In this algorithm, the state of the receiver remains constant or is frozen during a silent period. Once the receiver recognizes that a silent period has begun at the source, it inhibits the sampling clock pulses to the decoder during this silent period. This enables the decoder to remain at the same state until a new packet is received. The encoder, however, is changing its state continuously. Thus, the state of the decoder is different from that of the encoder when a new valid packet arrives, however, this will be eventually corrected by the error correction logic described earlier (Eq.(4)).

The main disadvantage of a freeze out is the presence of a large step size error ( the difference between the step sizes of the encoder and the decoder ) which requires several sampling periods for correction. The estimate error ( the difference between the estimates of the encoder and the decoder ) causes only a D.C. shift of the speech waveform.

2. Algorithm 2: Generation of a local periodic 11001100... steady state pattern at the receiver.

In this method, the receiver will locally generate a 11001100··· pattern at the input of the decoder during silent periods. This pattern enables the decoder estimate to leak to zero level. This is an acceptable pattern, since the encoder output has a 11001100··· when the input is in a silent period. However, because of the speech bits lost in silent packets, we still have a step size error. Finally during a silent period, the decoder is processing a local 11001100··· pattern which produces a periodic output whose fundamental frequency is equal to a fourth of the bit rate. This frequency is heard if the SVADM is operated at low bit rates.

3. Algorithm 3: Generation of a local periodic 101010··· steady state pattern at the receiver.

In this algorithm, the receiver will locally generate a 101010··· pattern instead of a 11001100··· pattern mentioned in algorithm 2. This pattern enables the decoder step size to become smaller. This causes a step size error. However, once the speech is initiated, the step size at the decoder grows larger and will approximately corrects itself due to the adaptive nature of the SVADM. In addition, there is also an estimate error which essentially causes a D.C. shift. It should be noted, however, that the D.C. shift in the estimate does not cause any problem as the human ear tends to ignore D.C. shifts. The advantage of this algorithm is that the periodic 101010··· pattern produces an estimate whose fundamental frequency is equal to a half of the bit rate and is not heard even at low bit rates unlike the one in algorithm 2.

## EXPERIMENTAL RESULTS

Figure 7 shows the test set up. It consists of a speech source, a band pass filter(B.P.F.), a DM encoder, a packetizer, a silence detector, a depacketizer, a steady state generator, a DM decoder, a B.P.F. and monitoring systems. The packetizer-silence detector and the depacketizer-steady state generator were simulated by a PDP-11/34 computer for real time operation.

For efficient silence detection using the output bits of the SVADM encoder requires an input noise voltage less than the minimum step size $S_o$ ($S_o$ = 10 mV). The speech source, which was used for the experiments, is a tape recorder. The noise voltage at the output of the tape recorder was less than 10 mV.

The parameters varied in the experiments were

(1) Packet size P, where P = 1024, 512 bits,
(2) Threshold $T_p$, where $T_p$ = 1/2, 1/4, 1/8, 1/16 and
(3) Samling rate $f_s$, where $f_s$ = 16, 9.6 Kb/s.

Experiment 1: Non-Repacking

We found by subjective comparison that there is a very little difference in the use of the three receiver compensation algorithms. In general, a local generation of a 1010··· at the receiver during silent periods was preferred for the reasons mentioned earlier. In addition, the subjective evaluation showed that the listeners of the processed speech found no recognizable degradation at $T_p$=1/2 and 1/4. However, at $T_p$= 1/8, they were able to distinguish the breaks in the speech. This was due to the fact that at lower thresholds more silent packets are not transmitted. Also, at $T_p$ = 1/16, the processed voice loses intelligibility.

We computed the effective packet rate of transmission ($r_e$) by keeping track of the total number of packets ($N_p$) assembled and the total number of silent packets ($S_p$) over a fixed period of time. The total time taken to transmit the packet is given by

$$T = (N_p)\ (P)\ /\ f_s \qquad (9)$$

where $f_s$ is the bit rate.

The effective packet transmission rate is given by

$$r_e = \frac{(N_p - S_p)}{T} \qquad (10)$$

Figure 8 shows the plot of $r_e$ as a function of $T_p$ for $f_s$=16Kb/s. At $T_p$ = 1/2, $r_e \approx$ 14 packets/sec, at $T_p$ = 1/4, $r_e \approx$ 13 packets/sec. and at $T_p$ = 1/8, $r_e \approx$ 12 packets/sec. For all the three values of $T_p$, the processed speech is intelligible. By detecting silence, the effective packet rate, $r_e$, is reduced. For example, at $f_s$ = 16Kb/s and P = 1024 bits, $r_e$ is approximately 16 packets/sec., when all the packets are transmitted. However, by detecting the silence periods, we obtain a reduction in the value of $r_e$. Thus at $T_p$ = 1/8, $r_e \approx$ 12 packets/sec. constitutes a reduction of 25% while still maintaining intelligible speech. This is a substantial reduction since the speech tape used had silent periods of approximately 25%, which was measured experimentally.

Experiment 2: Repacking and generation of a local 11001100··· pattern at the receiver, when a silent period is detected.

The use of the repacking and the introduction of a 11001100··· pattern at the receiver, during silence. improved the subjective quality of the processed speech at $T_p = 1/8$. The noticeable breaks in speech heard in experiment 1, were not present.

Here also, we computed the effective packet rate of transmission by processing the speech over a fixed period of time. In this experiment, we measured the total time (t) of speech processing. $r_e$ is, then given by

$$r_e = \frac{N_p - S_p}{t} \qquad (11)$$

Table 2 illustrates the computation of $r_e$ for different values of $T_p$. Figure 8 shows the plot of $r_e$ as a function of $T_p$. We notice that the values of $r_e$ are similar to non-repacking scheme. Thus, $r_e$ is still reduced compared to transmitting all the packets.

The periodic pattern of 11001100··· at the input of the decoder produces a periodic estimate whose fundamental frequency is equal to a fourth of the bit rate. When $f_s$ < 16Kb/s, this frequency is less than 4 KHz. This unwanted component can be heard at the output. In the next experiment, we overcome this problem by feeding a 101010··· instead of a 11001100··· to the SVADM decoder.

Experiment 3: Repacking and generation of a local 101010··· pattern at the receiver when silence is detected.

The use of a 101010··· pattern at the decoder input generates a tone at $f_s/2$ and is not heard. The subjective evaluation showed this scheme performed with approximately the same quality as that of experiment 2, with respect to speech intelligibility.

In Table 3, we have tabulated a subjective comparison of the non-repacking and the repacking schemes. The results of the

experiments 2 and 3 are combined. The two criteria, we use, for subjective comparison are (a) intelligibility and (b) acceptability. Intelligibility is self explanatory. Acceptability is best explained by an example or two. One is " The cat is brown ". The reproduced speech may contain " The cat brown ". In this instance, the received words are intelligible, but the syntax is lost. Thus, this is an unacceptable output. The second sentence is " His work is irrelevent ". The reproduced speech may contain " His work is relevent ". Here, we lose the first syllable of the last word and reach a wrong conclusion. Thus this output also is an unacceptable one. At $T_p$ = 1/2 and 1/4, there is no difference in the performance using the repacking and the non repacking schemes. At $T_p$ = 1/8, the repacking scheme enhances the quality of the processed speech significantly. However, at $T_p$ = 1/16 neither system is acceptable.

## CONCLUSIONS

Silence detection has been accomplished digitally by using the periodic steady state output of the delta modulator encoder. It has been established that by not transmitting the packets during silent periods of speech, the packet voice network can be built more efficiently, since there will be a decrease in the overall packet transmission rate without loss of speech quality. For lower threshold levels, the repacking scheme vastly increases the intelligibility of the processed speech.

REFERENCES

1. C.L.Song, J.Garodnick, D.L.Schilling, " A variable step size robust delta modulator ", IEEE Trans.,COM-TECH Vol. 19, pp 1033-1040, Dec. 1971.

2. C.J.Weinstein, " Fractional speech loss and talker activity model for TASI and for packet switched speech ", IEEE Trans., COM-TECH Vol. 26, Aug. 1978.

3. Motorola Incorporated, " Semi-conductor data library ", Vol. 6/ series B, pp 130-134.

4. Antoine M.Jousset, " Plans and principles for public data switched networks in France ", IEEE Proc., Vol. 60, pp 1370-1374, Nov. 1972.

5. S.M.Ornstein, F.E.Heart, W.R.Crowther, H.K.Rising, S.B.Russell, A.Michel, " The terminal IMP for the ARPA Computer network ", Joint Computer Conf., AFIPS Conf. proc., Vol. 40, pp 243-254, 1972.

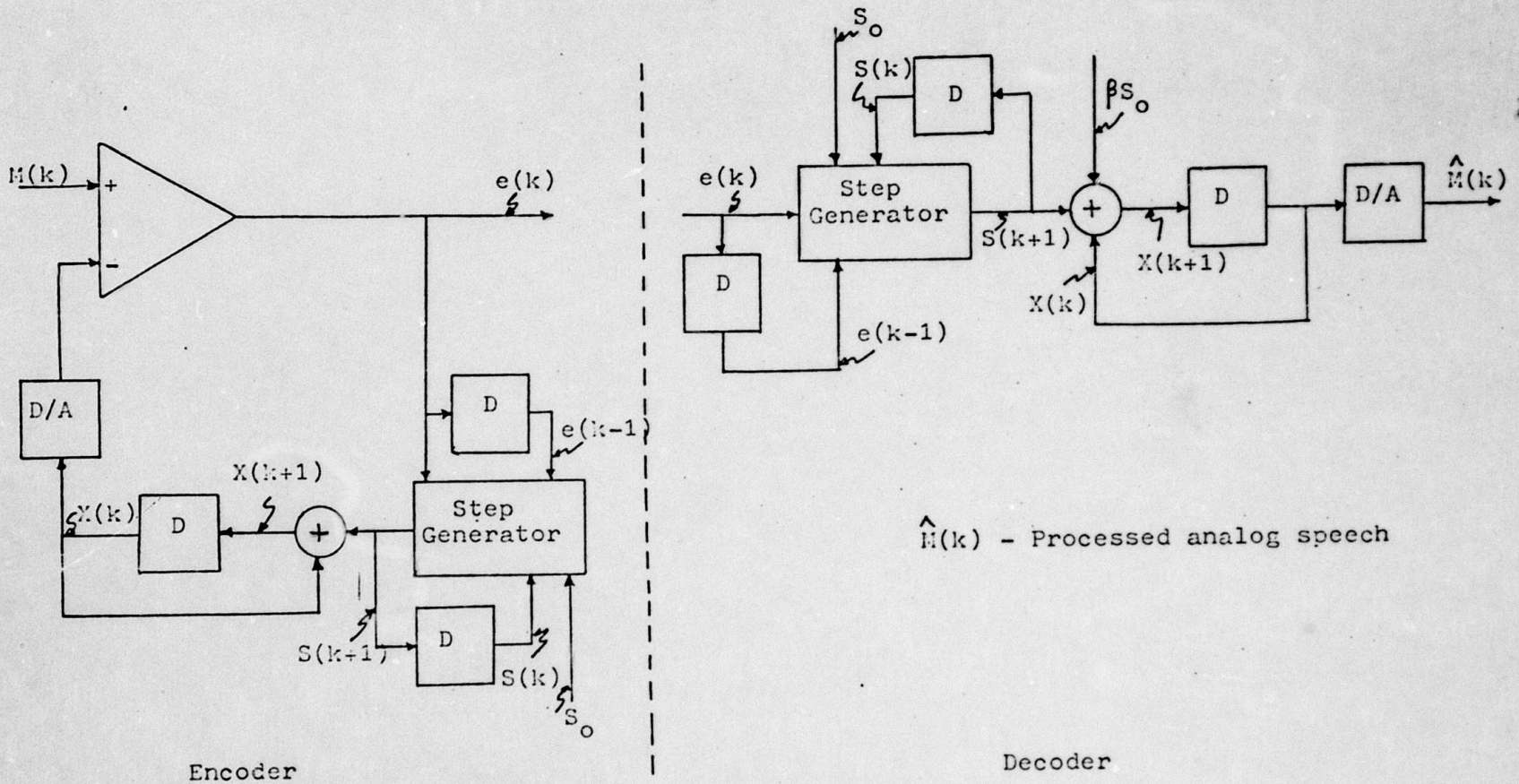



Fig. 1 Block diagram of the SVADM

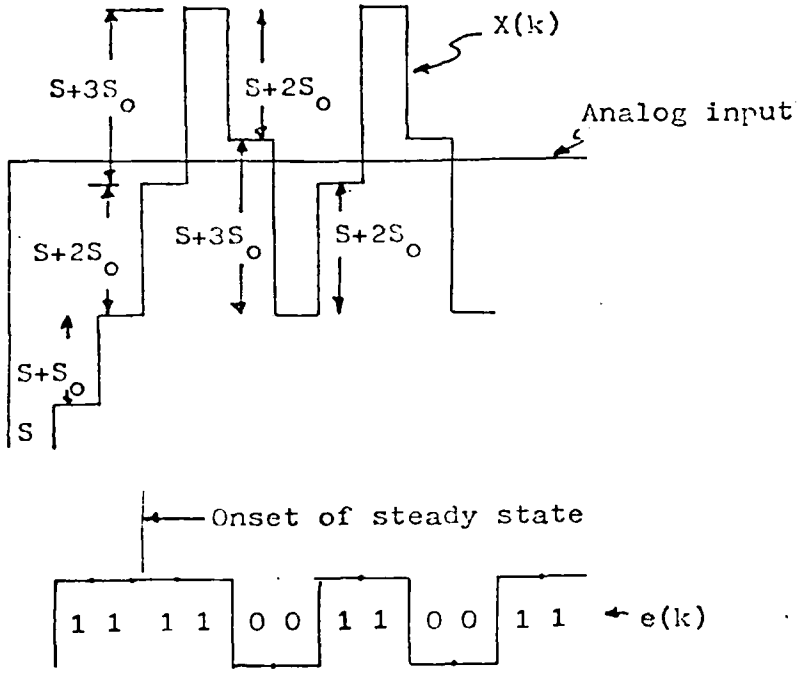



Fig. 2 Response of the SVADM for a step input.

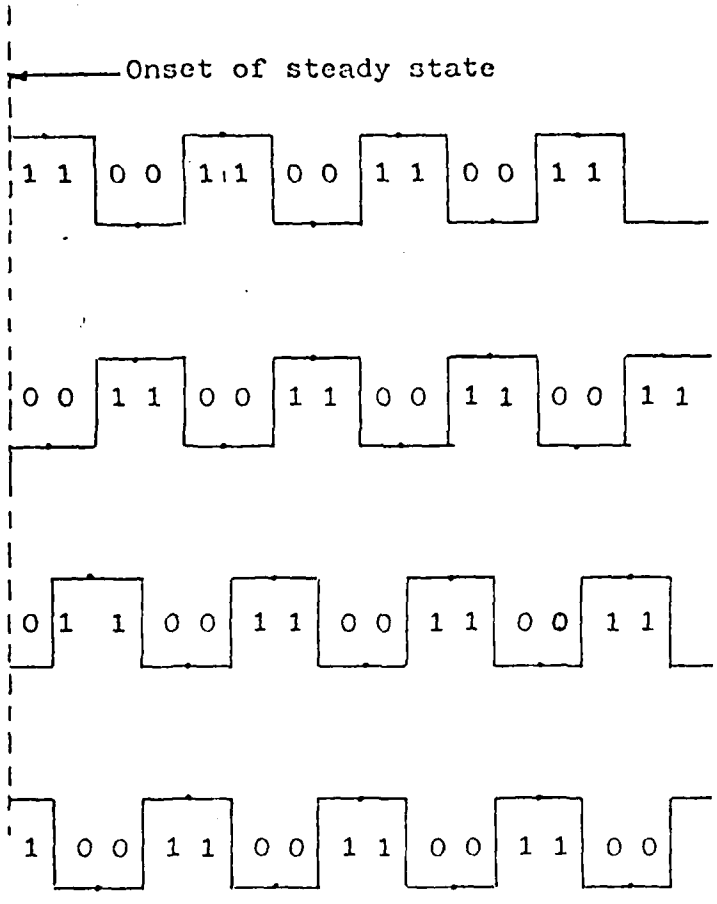



Fig. 3 The four steady state responses of the SVADM

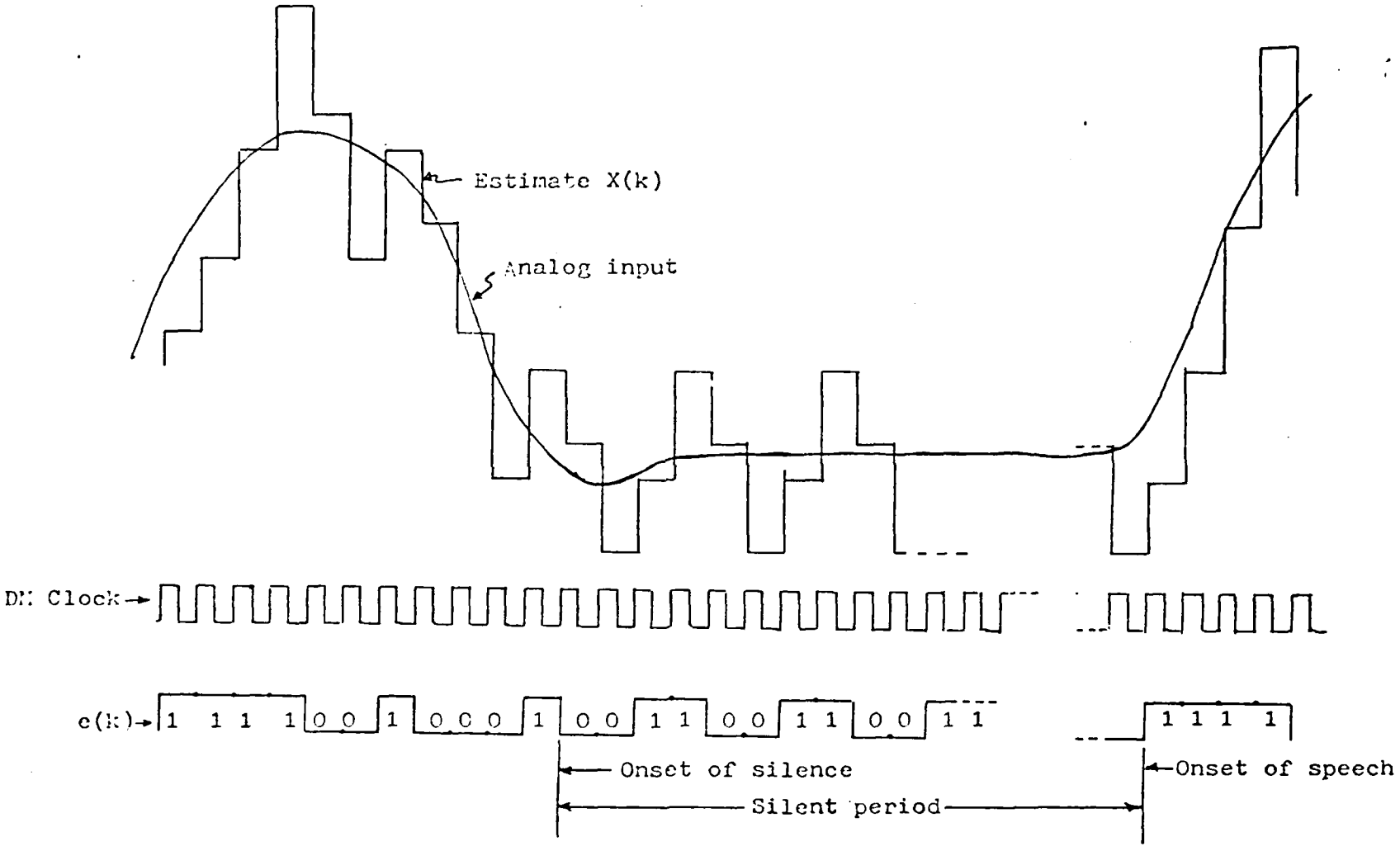
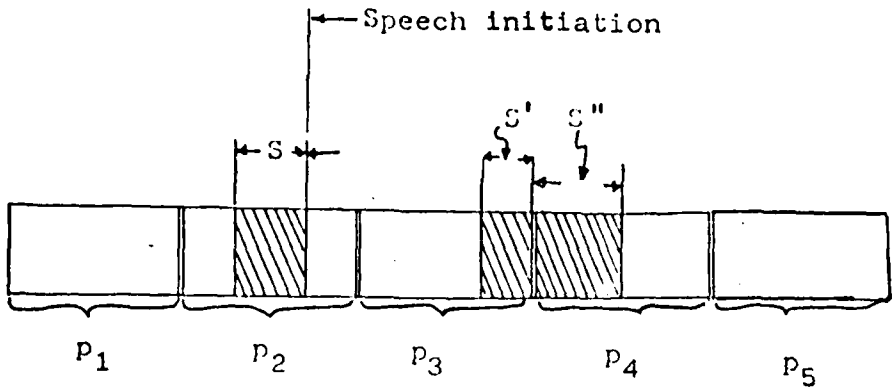



Fig. 4 Timing diagram for the onset of speech and the onset of silence




$S/P \geq T_p$ $S'/P < T_p$ $S''/P \geq T_p$

P - packet size of each packet.

Fig. 5 Determination of a silent packet

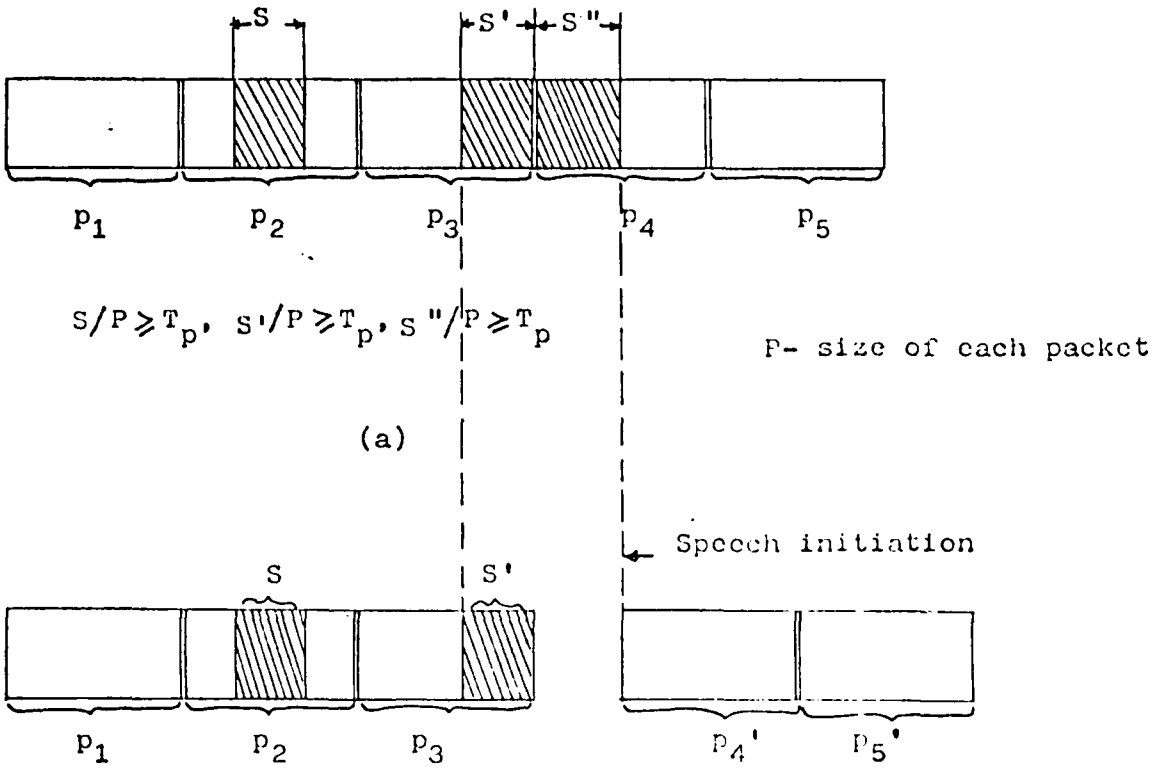
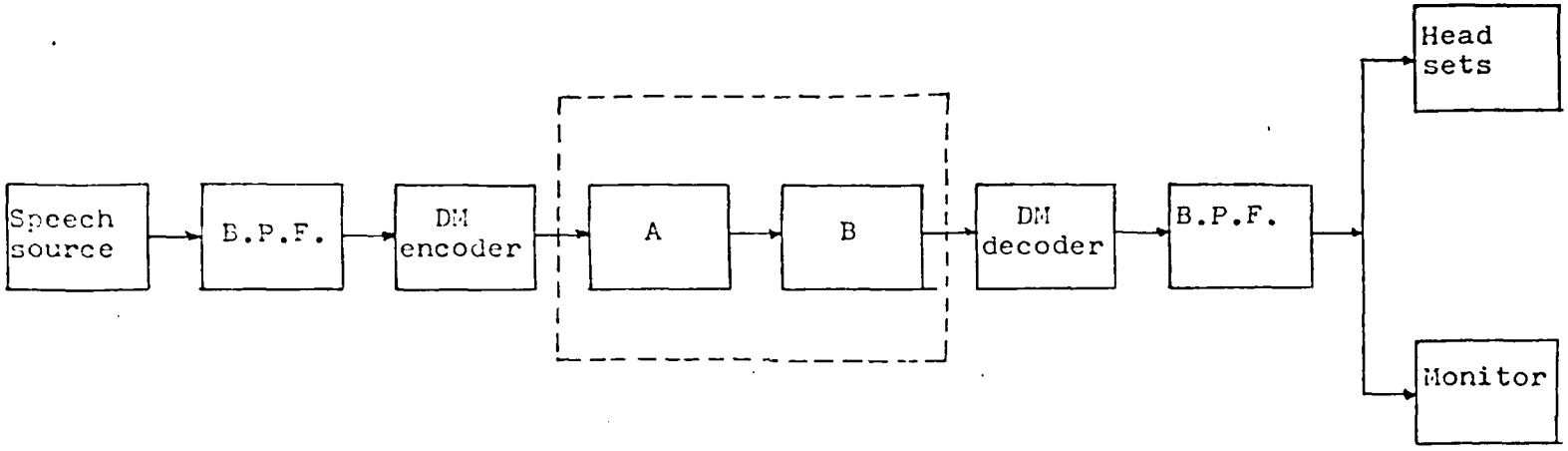



Fig. 6 (a) Non-repacking.
(b) Repacking.




A - Packetizer and silence detector

B - Depacketizer and steady state generator

B.P.F. - band pass filter set from 300 Hz to 2500 Hz

Fig. 7 Test set up for silence detection and speech onset detection

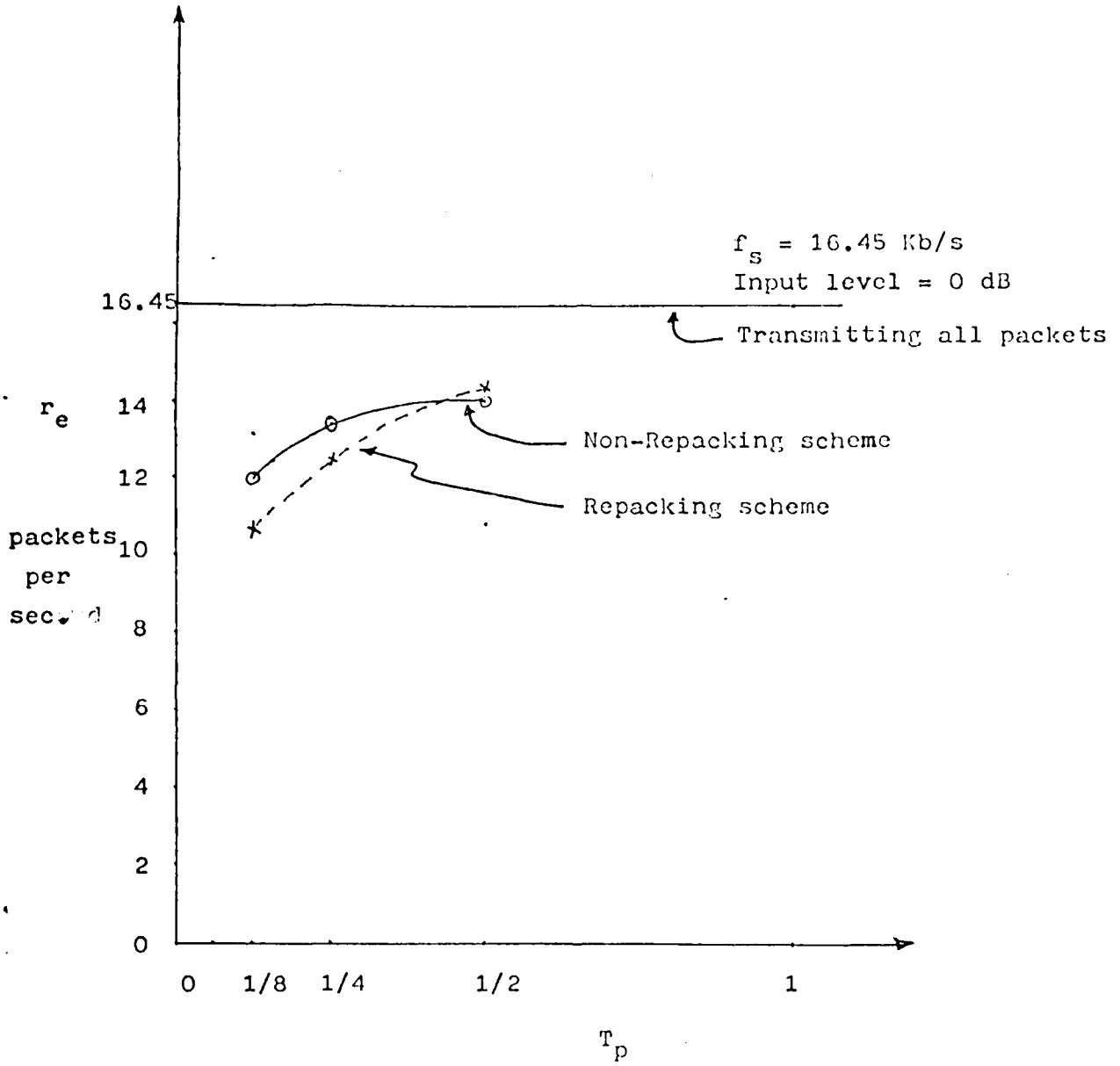



Fig. 8 Effective packet rate of transmission, $r_e$, as a function of $T_p$

Table 1: Computation of the effective packet rate of transmission. for " Non-Repacking " scheme.

Input level = 0 dB
Packet size = 1024 bits
Bit rate $f_s$ = 16.452 Kb/s

| | Threshold $T_p$ | | |
|---|---|---|---|
| | 1/2 | 1/4 | 1/8 |
| Total number of Packets formed, $N_p$ | 19911 | 16871 | 14747 |
| Total number of Silent packets, $S_p$ | 2654 | 2762 | 3703 |
| Total number of packets transmitted ( $N_p - S_p$ ) | 17255 | 14109 | 11044 |
| Total time taken to transmit the packets, $(N_p)$ $(P)/f_s$ secs. | 1239 | 1050 | 917.87 |
| Effective packet rate of transmission, $r_e$ | 13.9 | 13.4 | 12 |

Table 2: Computation of the effective packet rate of transmission for " Repacking " scheme.

Input level = 0 dB
Packet size = 1024 bits
Bit rate $f_s$ = 16.452 Kb/s
Total time of speech processing = 600 sec.

| | Threshold $T_p$ | | |
|---|---|---|---|
| | 1/2 | 1/4 | 1/8 |
| Total number of packets formed, $N_p$ | 9492 | 9561 | 9515 |
| Total number of silent packets, $S_p$ | 1002 | 2071 | 3150 |
| Total number of packets transmitted, $N_p - S_p$ | 8490 | 7490 | 6365 |
| Effective packet rate of transmission, $r_e$ | 14.15 | 12.4 | 10.6 |

Table 3: Subjective comparison of"Non-repacking" and "Repacking" schemes.

| Input level dB | $f_s$ Kb/s | P bits | $T_p$ | Non-repacking | Repacking | Comparison |
|---|---|---|---|---|---|---|
| 0 | 16 | 1024 | 1/2 | Intelligible. Acceptable. | Intelligible. Acceptable. | No difference. |
| | | | 1/4 | Intelligible. Acceptable. | Intelligible. Acceptable. | No difference. |
| | | | 1/8 | Intelligible. Not acceptable: speech is clipped. | Intelligible. Acceptable. | Repacking is significantly better than non-repacking. |
| | | | 1/16 | Not intelligible. Not acceptable. | Not intelligible. Not acceptable. | Words are missing. |
| -10 | 16 | 1024 | 1/2 | Intelligible. Acceptable. | Intelligible. Acceptable. | No difference. |
| | | | 1/4 | Intelligible. Acceptable. | Intelligible. Acceptable. | No difference. |
| | | | 1/8 | Not intelligible. Not acceptable. | Intelligible Acceptable: Breaks are noticed. | Repacking is preferred. |

Table 3: continued.

| Input level dB | $f_s$ Kb/s | P bits | $T_p$ | Non-repacking | Repacking | Comparison |
|---|---|---|---|---|---|---|
| 0 | 9.6 | 1024 | 1/2 | Intelligible. Acceptable. | Intelligible. Acceptable. | No difference. |
| | | | 1/4 | Not intelligible. Not acceptable. | Not intelligible. Not acceptable. | Words are clipped. |
| -10 | 9.6 | 1024 | 1/2 | Intelligible. Acceptable: Breaks are noticeable. | Intelligible. Acceptable: Breaks are less than that of non-repacking. | Repacking is preferred. |
| | | | 1/4 | Not intelligible. Not acceptable. | Not intelligible. Not acceptable. | |

## 1.5 Design of a Packet Voice Transmission System

This section describes the design of a packet voice network and the results of the evaluation tests performed. The packet voice network was simulated on a PDP-11/34 computer for real time operation. Adaptive delta modulators were used as source encoders. The average packet transmission rate and the subjective quality of the processed speech are presented.

### Introduction

As the development of computer networks proceeds, the need for voice transmission facilities over packet switched networks has been growing, especially for use in teleconferencing which is a natural communication tool between people. Up to this date a network voice protocol has been developed for the ARPA network and some measurements have been performed to determine the delay time distribution of packets. Similar research has been performed on several other networks.

It is well known that conversation becomes difficult if the round trip delay is greater than a few hundred milliseconds. In large packet switched networks, such as the ARPA network, the round trip delay can easily be greater than hundreds of ms, especially when the number of hops and the packet rate become large. Moreover, the delay time changes greatly from packet to packet. Researchers in ISI[1] showed that the average delay time as well as the variance becomes large if the packet rate exceeds 10 packets/sec on the ARPANET. In addition, the packet arrival sequence may be different from that transmitted. To cope with this situation, every packet is assigned a time stamp which designates the output time of the packet (network voice protocol). To resequence the packets, using the time stamps, requires the use of buffers at the receiving end. This increases

the average delay time of the packet leading to the degradation of conversational quality. As for the packet error, (the probability that some erroneous packets are received) it is relatively small because of error control which is usually used between adjacent switching nodes.

In this study, we evaluated the conversational speech quality in a situation where the round trip delay can change greatly, and we propose the design of a packet voice transmission system. We have simulated a real time packet voice transmission system and performed certain evaluation tests to determine the quality of the processed speech. The parameters used in these tests are delay time distribution, packet loss rate and silence detection algorithm. We have used the Song Voice Adaptive Delta Modulator (SVADM) at the source encoder.

## Packet Voice Transmission System

The system diagram of a generalized packet voice transmission system is shown in Fig.1. The voice waveform signal is encoded into a binary sequence and fed into the packetizer. The packetizer examines the bit stream, detects the start and the end of speech, packs the bits and makes up a sequence of packets. At the same time, it assigns the time stamp to each packet whose value designates the starting time of the packet. Packets which are generated by the packetizer are passed to the packet switched network in which every packet is delayed randomly and discarded with some probability (which simulates packet loss probability), and finally delivered to the receiver. A sequence regenerator buffers the packets, checks the value of time stamps with the present time, and makes up the output bit stream.

## Voice/Silence Detection Scheme in Packetizer

Although the speech waveform is transmitted in a digital

format, the bit stream during silent periods is neglected. Consequently, the voice/silence detection scheme plays an important role in reducing the effective packet rate. The detection method used is shown in Fig.2. The input bit stream is processed in groups of 16 bit words. Every incoming word is stored in a shift register whose word size is fixed. It is then compared with several fixed bit patterns which are the typical bit streams at silent periods, and the result (match or no match) is stored in another shift resigter of entry length $L_{max}$ . After that, the total number of matches in this register is compared with some constant whose optimal value is dependent on the present input mode.

When in the silent mode, the number of matches in the shift register is compared with a constant $V_o$. If the number is less than $V_o$, the start of the active speech is detected and packetization begins. At the head of the first packet, a number of the previously stored words (pre-offset) is inserted to preserve the start of speech. In the voice mode, the number of matches is compared with a constant $S_o$. If the number is greater, the end of speech (silence) is detected. At that time, some input words (post-offset) previously stored in the packet buffer are discarded to shorten the packet length.

Sequence Regeneration Scheme

The delay time of each packet through the network varies from packet to packet. Therefore, the order of received packets does not always match the order of those transmitted. Furthermore, the packet location on the time axis may fluctuate from the original. When the variance becomes large, we cannot neglect its effect on the quality of speech. The limit of the variance for which we do not need any form of sequence regeneration is fixed by subjective evaluation of conversational speech quality.

When the variance is greater than the limit, the use of a sequence regeneration scheme is unavoidable. The scheme which we propose is as follows:

Let us assume the delay time distribution is as in Fig.3. Packets with delay time less than $T_s$ are stored in buffers: those with delay time greater than $T_s$ are discarded. $T_s$ is the absolute constant delay time of the packets between the source encoder and the destination decoder. At time $T_s$ stored packets are outputted to the decoder.

The real shape of the delay distribution curve is shown to be similar to Fig.3[2], with most of the delay time concentrated near the minimum. Although the probability of occurence of large $T_s$ is rather small, the distribution spreads to the very large delay time region. If $P_e$ is to be very small, $T_s$ can become sufficiently large so that the round trip delay becomes intolerable. $P_e$, which is the probability that the delay time is greater than $T_s$, gives the effective packet loss probability due to long delay time.

## The Simulator

The block diagram of the packet voice transmission simulator is shown in Fig.4. The functions of packetizer, packet network and sequence regenerator are all performed by the PDP 11/34 computer. This simulator has been used for real time system evaluation.

## Hardware Configuration

A PDP 11/34 minicomputer was used along with a DR-11 digital input/output interface to connect external devices to the computer. The specification of the control device used as interface (using 28 TTL Logic I.C.'s), between

the DR-11 and a pair of encoder/decoder is as follows:

16 bit parallel input/output to/from computer for each channel.

16 bit parallel to/from serial conversion

Bit streams from both encoders are stored bit by bit in shift registers (16 bit words), parallel transferred to the input buffer of the Dr-11 and read into the computer memory. As the same clock is supplied to both encoders, input data for each channel is made up at the same time and read into memory sequentially. Data is read out of the computer after every read-in operation. From the output buffer of the DR-11 two words are placed into shift registers, one word for each channel, and continuous bit streams are generated for the decoders of both channels.

Software Configuration

The operation of the simulator program is shown in Fig.5. The input/output processes are shown in Figs. 6(a) and 6(b) respectively. The program consists of 300 machine language instructions. The data area comprises 4K bytes (256 blocks) of packet buffer control blocks, and 16K bytes of packet buffer area for each channel, making up 36K bytes in total. After the read/write operation, the processing is performed sequentially for each channel. The processing sequence for each packet is as follows:

1. Voice detection (if in silence mode)
2. Allocation of packet buffer
3. Random Delay time generation
4. Insertion of packet buffer into the proper location of output-packet chain
5. Word collection

6. Silence detection (if in voice mode )
7. Comparison of the assigned output-time with present time and decision to output
8. Outputting of either words from packet buffer or silence patterns

To perform these tasks we use 3 packet buffer chains. A new packet buffer is acquired from the idle buffer chain, and an incoming word is stored in the buffer. The packets in the output chain are stamped with the output time and arranged in increasing order for transmission. If a new packet is created and the output time is assigned, the packet should be inserted into the proper location in the output packet chain by searching the chain. Process No.4 (above) requires considerable processing time. For example, the number of packet buffers which exist in the computer can be greater than 40 in some cases. The margin which is permitted in each cycle for word processing is limited. 'Cycle' is the time unit from an input of a channel to the next input of the same channel. All time values are normalized to this unit. Processes No.3 and No.4, which are done at the time of new packet creation are time - divided into several sequential tasks, each of which is executed within a single word processing cycle. If N cycles of search operation are required to find the location, N+3 cycles in total are needed to complete the processing.

Output Time Generation for Each Packet

The arrival time of each packet can be calculated as follows:

$$T_{arv} = T_{create} + T_{min} + T_{random} \qquad (5)$$

where $T_{create}$ is the time when the packet is created,

$T_{min}$ is the minimum delay time of the packet switched network, and $T_{random}$ is a random delay time. For the distribution function of $T_{random}$, 2 kinds of functions were assumed.

1. Flat density function
2. Approximate function of the measure result for the ARPA Network [ 2].

Random number generation was realized by

$$X=C \cdot X \qquad (6)$$

where C=37, and X is a 16 bit integer.

System Evaluation

The system has been evaluated by conducting the following tests:

Variation of Parameters in Silence/Speech Detection

Some of the important parameters such as the average number of transmitted packets and speech quality have been obtained by varying the parameters used in the silence/speech detection. Results appear in Fig. 7(a), (b), (c). In addition, packet size distribution measurements show that more than 95% of the packets are of full size. Speech quality was categorized in the following way:

Excellent - not different from or better than (due to silence rejection) the original speech.

Very Good - slightly different from original with no chopping of voice.

Good - slight degradation of speech due to chopping.

Fair - continuous chopping of voice although speech is still intelligible

Poor - unintelligible

Subjective Evaluation of a Two-Way Conversation With Constant Network Delay

With the parameters for silence/speech detection set

at the optimal and packet size of 128 bites, the ease with which a two-way conversation can be carried out has been evaluated. This test is conducted with a fixed time delay introduced in the system. The subjects are asked to rate the system into various categories as indicated in Table 1, as follows:

Very Easy - not different from local telephone.

Easy - conversation manageable with time needed for adjustment.

Difficult - difficulty in conversing due to large round trip delay.

Network Performance as a Function of Packet Loss and Random Delay

The quality of speech, introducing probabalistic packet loss and random delay time (random arrival) with flat distribution from $T_{min}$ to $T_{max}$ has also been obtained. Results are available in Fig.8.

System Design Methology

As a result of the delay time distribution and packet loss probability measurements a packet voice transmission can be designed. From these values the optimal system parameters for the speech/silence detection scheme can be obtained.

The number of words reserved for future speech/silence decisions should correspond to from 10 to 30 ms of speech. If we use 16K bits/sec. of delta modulation, $L_{max}$ must be greater than 30 words (30 ms). Therefore, 32 is selected as a good number for $L_{max}$. The optimal value of the pre-offset and the post-offset are 8 and 16 words respectively. Those for the threshold parameters $V_o$ and $S_o$, to change the processing mode, are 3 and 10 words.

Time Stamp Handling

If the absolute delay time is greater than 200 ms., we usually have difficulty with conversation. If the variance of the delay time exceeds 24 ms, we should be forced to use sequence regeneration scheme such as time stamping, when sequence regeneration is used it is suggested that the resulting constand delay time $T_s$ between encoder and decoder should be adjusted so that the probability of packet loss due to a large delay time becomes less than $10^{-2}$. After $T_s$ is fixed, the number of buffers needed for sequence regeneration can be calculated as follows:

$$N_b = F\ T_s/P \qquad (7)$$

where P is the average length of the packets in bits.

Conclusions

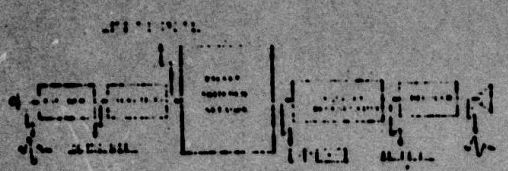
In the above discussion, we assumed that the network characteristics are fixed and can't be changed. As the development of packet transmission systems progresses, it is expected that packet networks will have packet voice capability. At that time, packet networks will be designed with the provision that 99% of the packets will have a coast to coast delay time less than 300 ms. With the progress of packet switching speeds, the average delay time induced by one packet switch can be less than 1 ms. Digital transmission bit rate of 10 Mbits, to connect packet switching facilities, may be reasonable in the future as well.

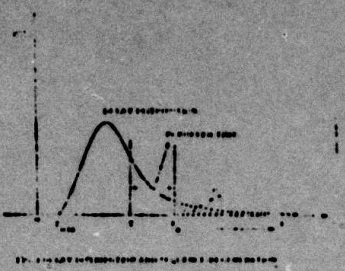
With coast to coast transmission delay of about 20 ms in case of terrestrial link, and 250ms in case of satellite, a packet network for voice, as well as data, transmission will be easily achievable.

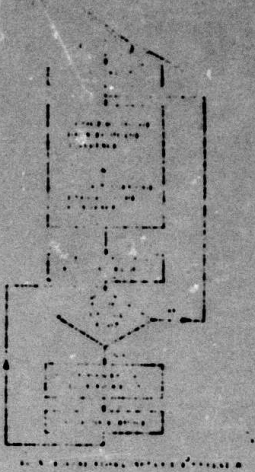
References

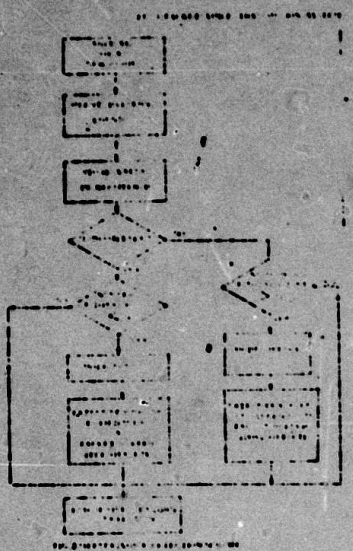
1) Cohen D., "Specifications for the Network Voice Protocol". NSC Note 68 (RFC741, N1C42444) Jan. 1976.

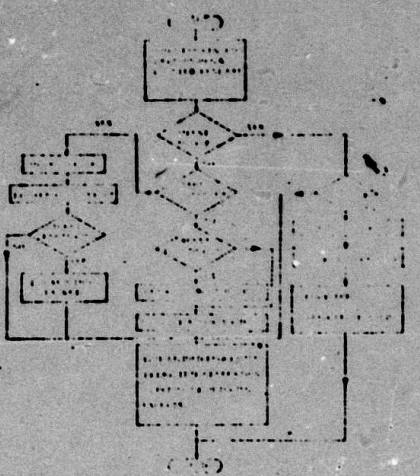
2) S.L. Cansey, E.R. Masler, E.R. Cole, "Some Initial Measurements of ARPANET Packet Voice Transmission" Conf. Rec. NTC '78. Birmingham, Alabama, pp 12.2.1-12 2-5, Dec. 1978

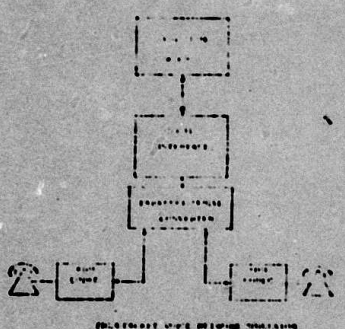


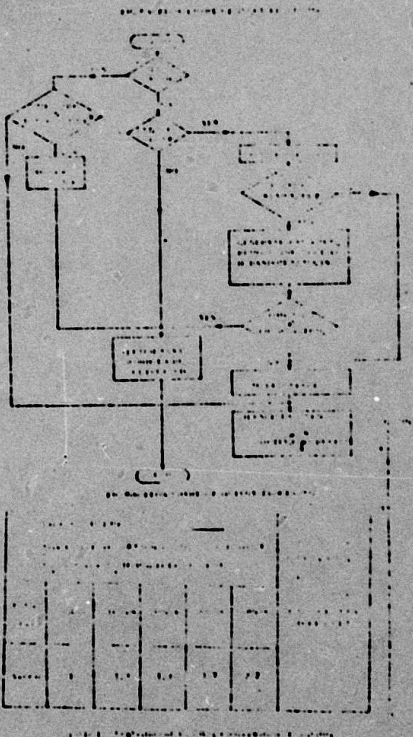


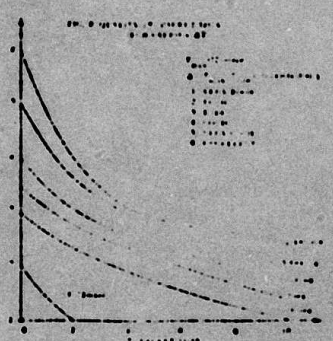












# CHAPTER II

## Video Encoding

### Introduction

A video signal typically has a bandwidth of 4MHz. In standard American television systems the picture content of the signal is presented on a raster of approximately 500 lines called a "frame" which is repeated 30 times/secs. Thus, the time that it takes to present each one of the 500 lines is approximately 1/15,000 sec. We say the "line rate" is 15,000 lines/sec and one can actually hear this signal if one stands near to the monitor. In actual practice the 500 lines are divided in half, the odd lines being presented during the first 1/60 sec and the even lines being presented during the next 1/60 second. This division of a 500 line frame into two interleaved 250-line "fields" is done so that the picture will have no perceptible flicker.

It is often convenient to digitally encode a video signal prior to transmission. This can be done using standard PCM techniques. Since the bandwidth of the signal is 4 MHz the Nyquist sampling rate of the system is 8M samples/s. The sampling rate is the rate of displaying picture elements and is often called the "pixel" rate or "pel" rate. The A/D converter in the PCM system encodes each sample into N bits. When N=8 the resulting picture quality is quite good, however, when N=6 the quality is significantly degraded. The transmitted bit rate for PCM is then between 6 x 8 = 48 Mb/s (6 bits/pixel) and 8 x 8 = 64 Mb/s (8 bits/pixel). In either case the bit rate is extremely high. A high bit rate requires a wide bandwidth for transmission; as a matter of fact the bandwidth is numerically equal to the bit rate. Another way of looking at the effect of high bandwidth is to note that a frame lasts 1/30 sec.

Thus to store a single frame of picture requires a memory size D of

$$\frac{48}{30} \simeq 1.6 \text{ M bits} < D < \frac{64}{30} \simeq 21 \text{ M bits}$$

As a result of this very large storage requirement PCM is usually not considered practical, and instead, other techniques are employed such as Transform Coding, Delta PCM (DPCM) or Adaptive Deltamodulation (ADM).

One transform coding technique called Hadamard transform coding has been studied extensively at Ames Research Center and has been shown to be able to encode pictures at a rate of 4 to 8 Mb/s (0.5 to 1 bit pixel). Thus, to store a single video frame of picture now requires a memory capacity of only

$$0.13 \text{ M bits} < D < 0.27 \text{ M bits}$$

Unfortunately this saving in bit rate is accomplished at the expense of hardware and computational complexity which makes the system somewhat undesirable.

The system suffers from an inherent weakness of this particular bandwidth reduction scheme: high sensitivity to errors. The system can trade error correction capabilities for redundancy but then the bit rate will increase. This problem makes Hadamard transform coding unsuitable for most applications.

Delta PCM has also been studied extensively. These systems operate at bit rates of 16-32 Mb/s (2-4 bits/pixel). A discussion of a DPCM system proposed for use on the space shuttle is contained in the IEEE Transactions on Communications, Nov., 1978, p.1671. It is seen that the delta modulator achieves comparable quality at a much lower cost, size, power consumption and at a much improved error sensitivity.

A DPCM system becomes badly degraded at a $10^{-4}$ error rate, while the ADM operates well at a $10^{-3}$ error rate, and is usable at a $10^{-2}$ error rate . Furthermore the DPCM systems proposed require a large number of IC's and the resulting power dissipation is very high by comparison.

The adaptive delta modulator is capable of encoding a video signal using bit rates of 8-16 Mb/s (1-2 bits/pixel). Thus, the memory capacity needed to store a frame of memory is now:

$$270 \text{ K bits} < D < 540 \text{ K bits}$$

While this storage is twice as large as the storage for the Hadamard encoder, the ADM system is smaller, more rugged and is much less costly. Furthermore the ADM retains the advantage of being extremely insensitive to errors caused by channel noise and operates well, even when the error rate is as high as $10^{-2}$ errors/bit.

In this chapter we discuss the use of the ADM algorithm developed by Schilling, Song and Garodnick (An ADM using this algorithm is commercially available from Deltamodulation Inc.), the block diagram of which is shown in Fig. 2-1. The equations of this ADM are

$$E_{k+1} = \text{sgn}\left[ S_{k+1} - X_k \right] \tag{2-1a}$$

$$Y_{k+1} = \begin{cases} | Y_k | \left[ E_{k+1} + 0.5\, E_k \right] & \text{if } | Y_k | \geq Y_{min} \\ Y_{min}\, E_{k+1} \end{cases} \tag{2-1b}$$

and

$$X_{k+1} = X_k + Y_{k+1} \tag{2-1c}$$

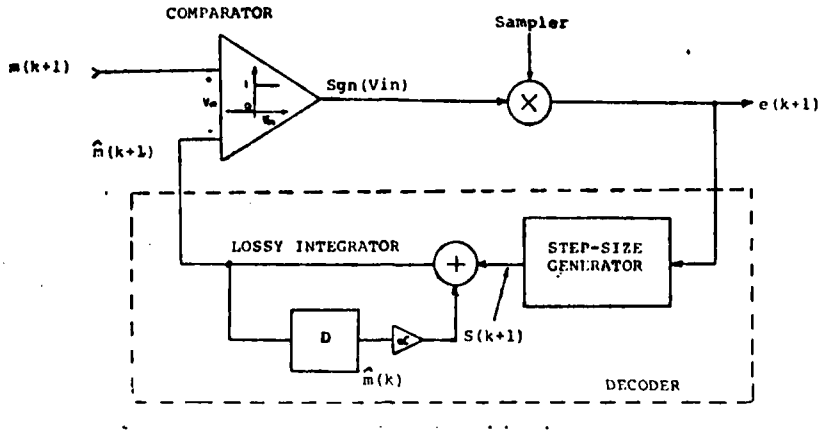



Fig. 2.1 Block Diagram and D-MOD

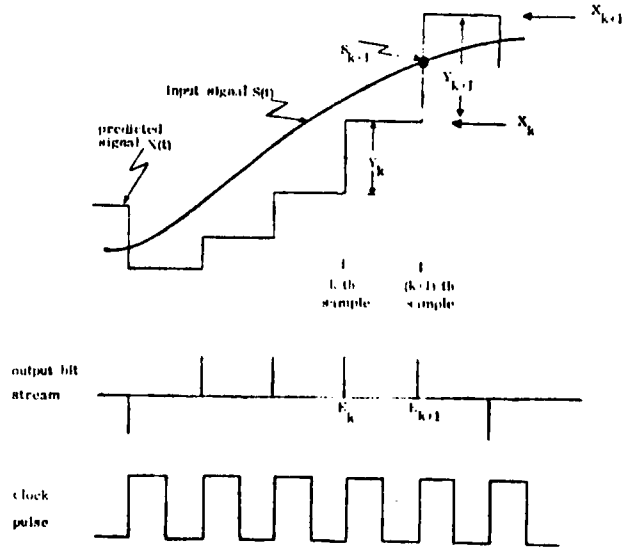



Fig. 2.2 D- MOD Clock, Estimate, Output, Input

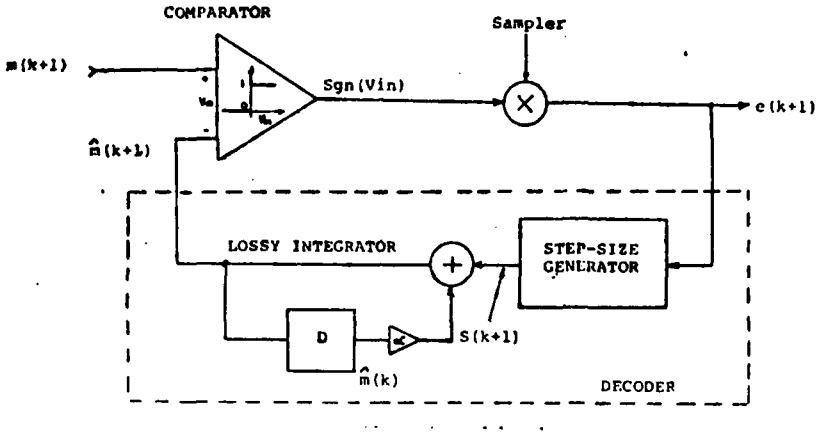



Fig. 2.1 Block Diagram and D-MOD

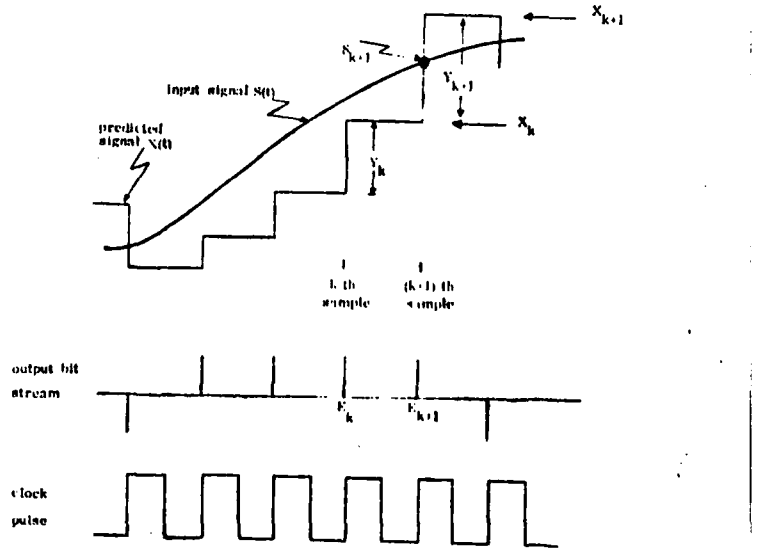



Fig. 2.2 D- MOD Clock, Estimate, Output, Input

where

$E_{k+1}$ is the transmitted bit
$S_{k+1}$ is the present sample of the input signal to the encoder
$Y_{k+1}$ is the step-size of the delta modulator
$Y_{min}$ is the minimum step-size
$X_{k+1}$ is the predicted value of the input sample

Figure 2-2 shows the relationship among the clock pulse, output bit stream, input signal and estimate. Observe that when the estimate $X_{k+1}$ is less than the sample of the input signal $S_{k+1}$ the transmitted bit is a "1" and the step-size is increased by the factor 1.5. Thus, the estimate rises exponentially, and can closely follow any rapid transition in gray level. When an overshoot occurs indicating that

$$X_{k-1} < S_k < x_k$$

the transmitted bit is a "0" and the step-size decreases by the factor of 0.5. The value 0.5 was chosen since, with equal likelihood, $S_k$ can lie anywhere between $X_{k-1}$ and $X_k$, thus with

$$X_k = X_{k-1} + Y_k$$

we set

$$X_{k+1} = X_k - 0.5\ Y_{k-1} + 0.5Y_k$$

That is $Y_{k+1}$ is chosen to be $0.5Y_k$ to place $X_{k+1}$ midway between $X_k$ and $X_{k-1}$.

## 2.1 Slow Scan Video Encoder/Decoder

There are many applications in which the video picture does not change for perhaps one minute or more. Such applications are in multimedia presentations, such as map viewing, teleconferencing, computer managed video communication, airline reservations, flight scheduling, etc. When the picture remains stationary for a long period of time, there is no need to continually transmit the redundant bits as it adds no information to the present signal. For example, we saw that using an ADM encoder, the number of bits that constitutes a complete frame, at a bit rate of 16 Mb/s is 540 Kbits. If these bits are transmitted at the normal rate of 30 frames/sec, we must transmit the data at the encoded bit rate of 16 Mb/s. However, if new information is provided at the rate of 1 frame each minute, the average bit rate is reduced to

$$\frac{540 \text{ Kbits}}{\text{frame}} \times \frac{1 \text{ frame}}{60 \text{ seconds}} = 9 \text{ Kbits/second}$$

a significantly reduced bit rate. If we assume that there are 1000 bits/packet the slow-scan packet rate is 9 packets/second which is less than the packet rate required to transmit voice.

As a matter of fact the data can be modulated by a modem for transmission using a telephone network. If, on the other hand, we were to use PCM encoding techniques a frame change could occur only after each 3-4 minutes. A second very practical consideration is that using delta modulation techniques we can eliminate the need for any word synchronizing circuitry.

### Frame Storage

There are two ways to store a frame of video signal: analog and digital.

In the analog system, the frame of signal is stored in a storage tube and, when required, slowly read out into the ADM encoder which can operate at the low rate of say 9 Kb/s. Thus, the same ADM could be used for voice and slow-scan video.

In the receiver the digital signal is received by the ADM decoder, converted to an analog signal and stored in a second analog storage tube. The output of this tube drives the TV system.

During the past few years almost all applications using image storage have changed from analog to digital devices. The problem with analog storage is that the system is large, costly and is of inferior quality. the analog storage device stores the image using surface charge concentration techniques. This provides marginal picture quality. System noise is found to increase with time, degrading the stored picture; also some leakage occurs. Both effects act together to produce a somewhat "washed-out" appearance to the picture.

Digital frame storage techniques are inexpensive, they do not require the periodic maintenance of the analog storage devices, and we will not observe any degradation of the S/N ratio or of any other aspect of the picture quality independently of the storage time. The S/N ratio of the stored image can be arbitrarily large and is determined by the digital encoder at the front end of the system. In our system we use an Adaptive deltamodulator. The ADM is a digital device, hence a frame of signal will be first encoded into a stream of digital signals and then stored in the digital memory. The bit stream to be transmitted is read out of memory at any, arbitrarily set, slow rate. This digital signal when received by the receiver is again stored in memory and is read out, into the ADM decoder, at the real time video rate. The analog output of the decoder

is then displayed on the monitor.

Since the application required the transmission of good quality, digitally encoded video, we decided to design a custom digital frame storage memory which would work in conjunction with a pair of ADM's at a bit rate of 16 Mb/s.

Real Time Digital Storage

A block diagram of the slow scan video encoding system is shown in Fig.2.1-1. Note that the camera signal inputs the ADM encoder which in turn drives the memory. In order to keep the cost of the system down we used relatively slow memories. Using a memory multiplexing technique, we are able to operate the memory at an apparent speed of 16 Mb/s even though the individual memories can only operate at a speed of 1 Mb/s. This memory multiplexing scheme is shown in Fig. 2.1-2. Here we see that the signal after being delta modulator encoded is put into a high speed (TTL) 16-stage serial/parallel converter. Each stage of the register is transferred to a 32 K x 16 MOS memory as shown. Thus, the writing speed into the MOS memory is 16K/16 = 1 Mb/s which is well within the ability of the MOS units. In order to provide for an arbitrary bit rate at the output we use a latch between the memory and the output parallel/serial converter. Thus the latch will always have available the next data while the present data is being shifted out at the slow rate.

System Design

The memory system design is shown in Fig. 2.1-3. Only one memory was constructed and synchronization was obtained by using the encoder clock to derive the decoder clock. The system was tested using test slides as well as moving pictures. The results, as expected, look identical to the results of a video ADM operating without the memory. However, now the signal can be recorded, held and played back at any desired time delay and rate.

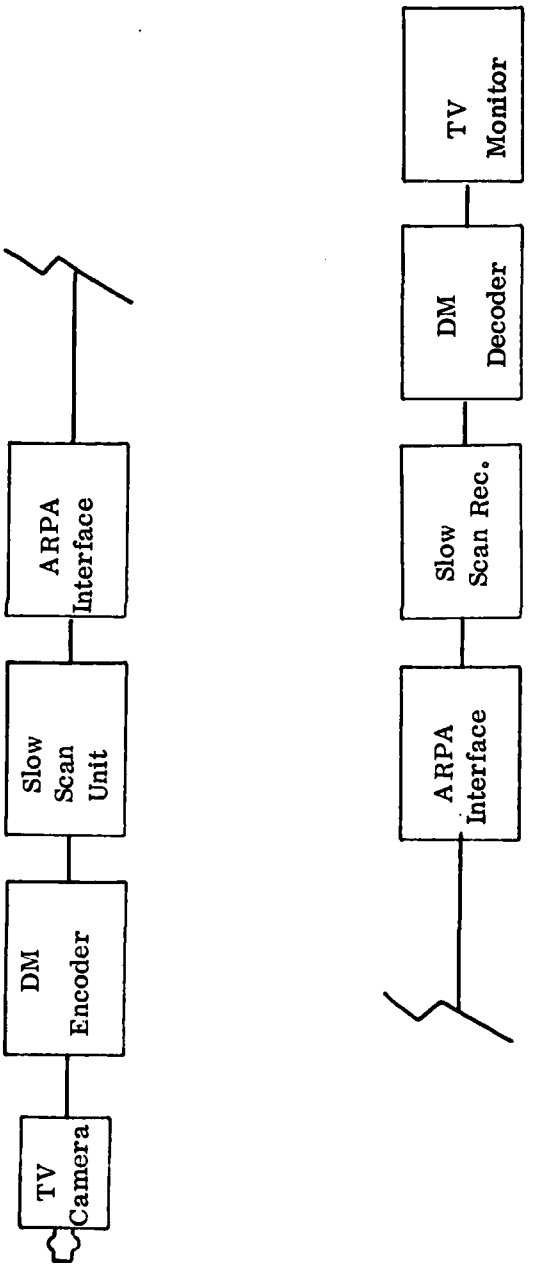



Fig. 2.1-1 Block Diagram of the Slow Scan Video System

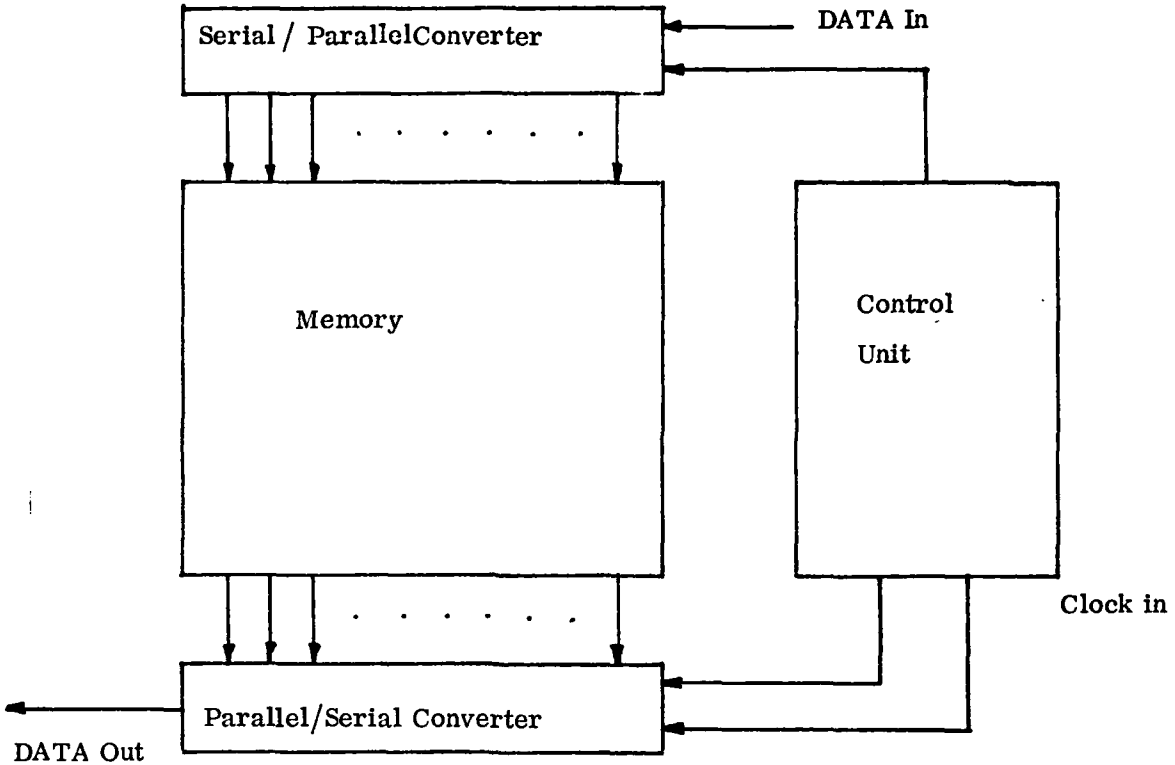



Fig. 2.1-2 Memory Multiplexing Technique

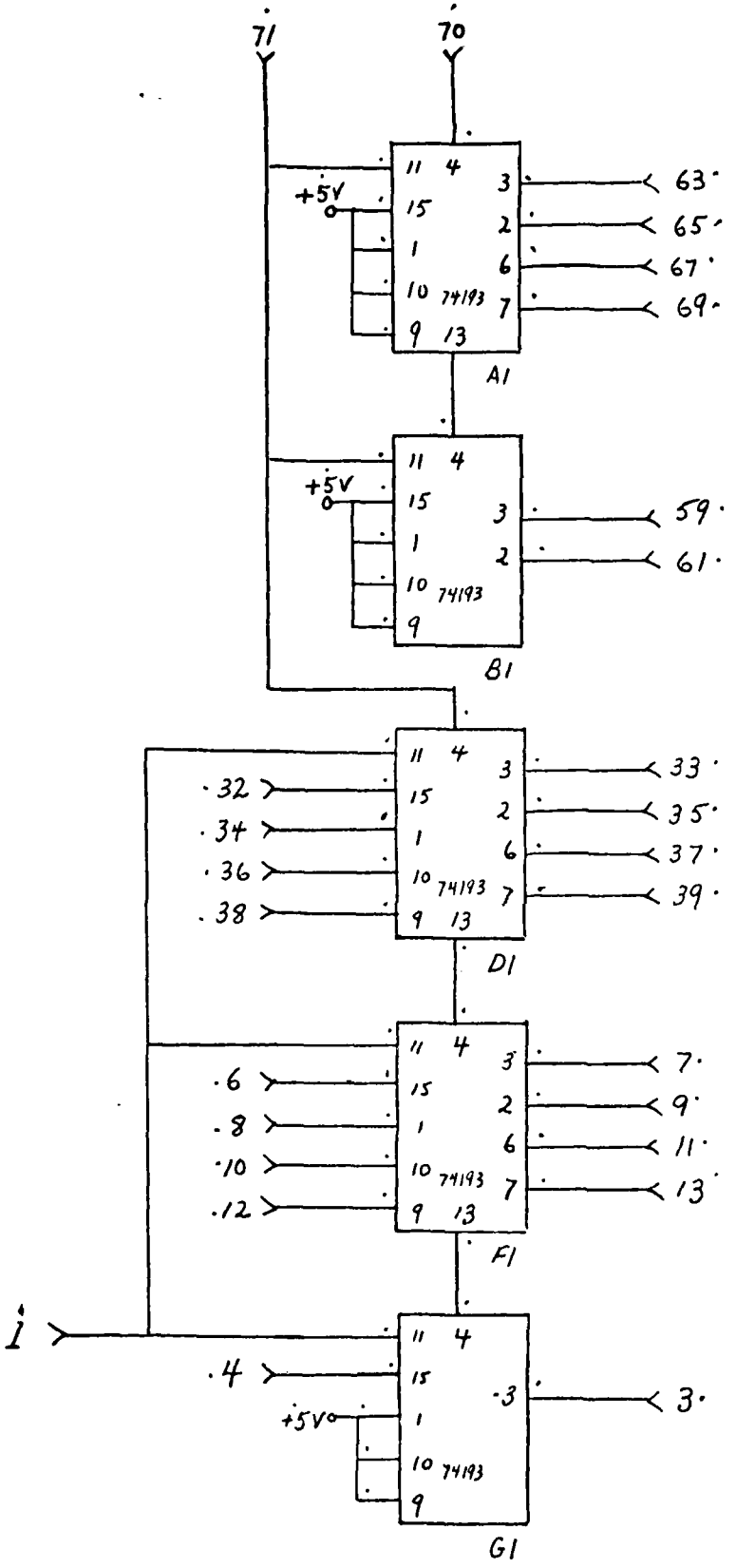



Fig. 2.1-3 Address Generator for Memory Unit

Fig. 2.1-3 Control Unit for Slow Scan

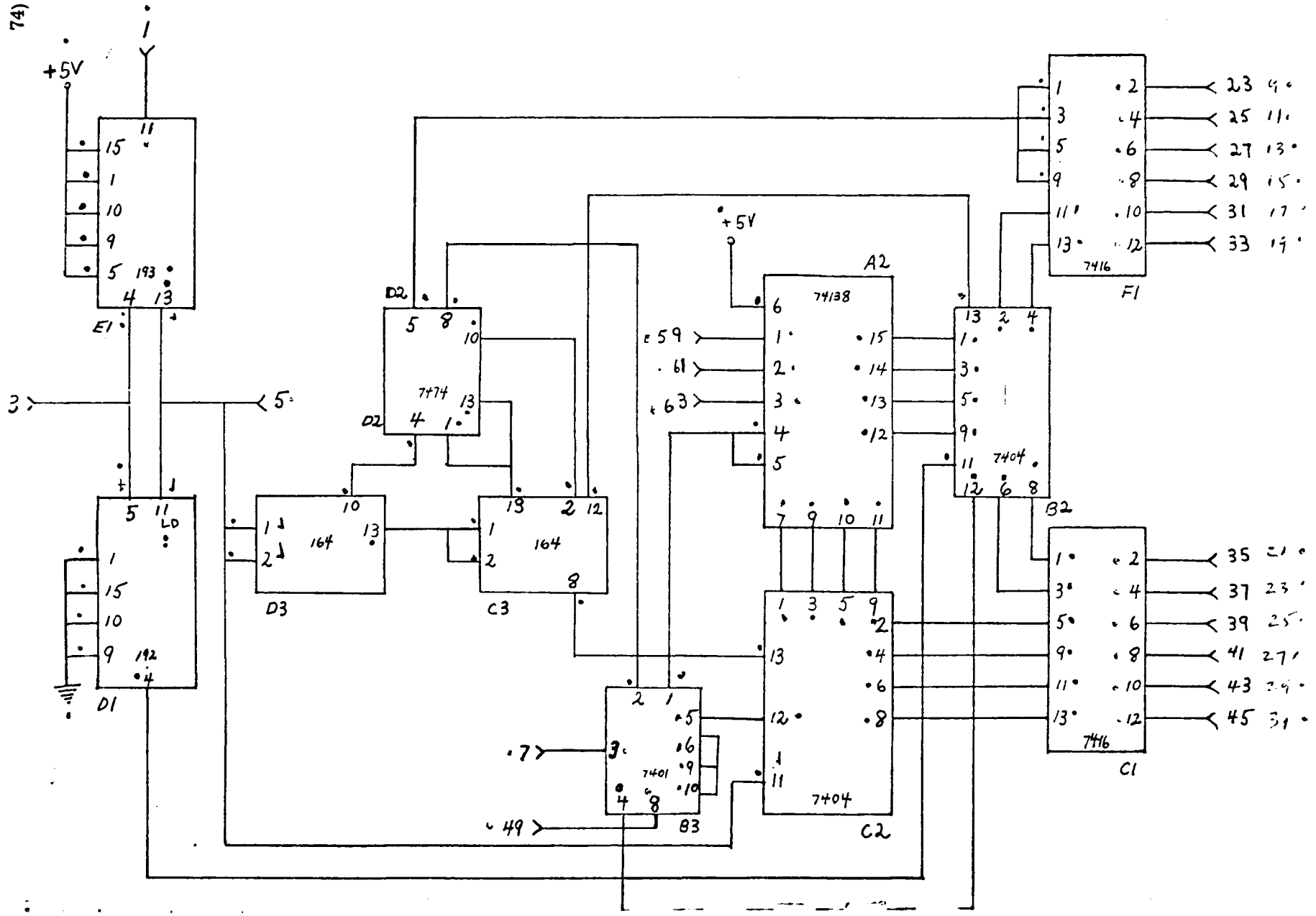


Fig. 2.1-3 Serial to Parallel Converter

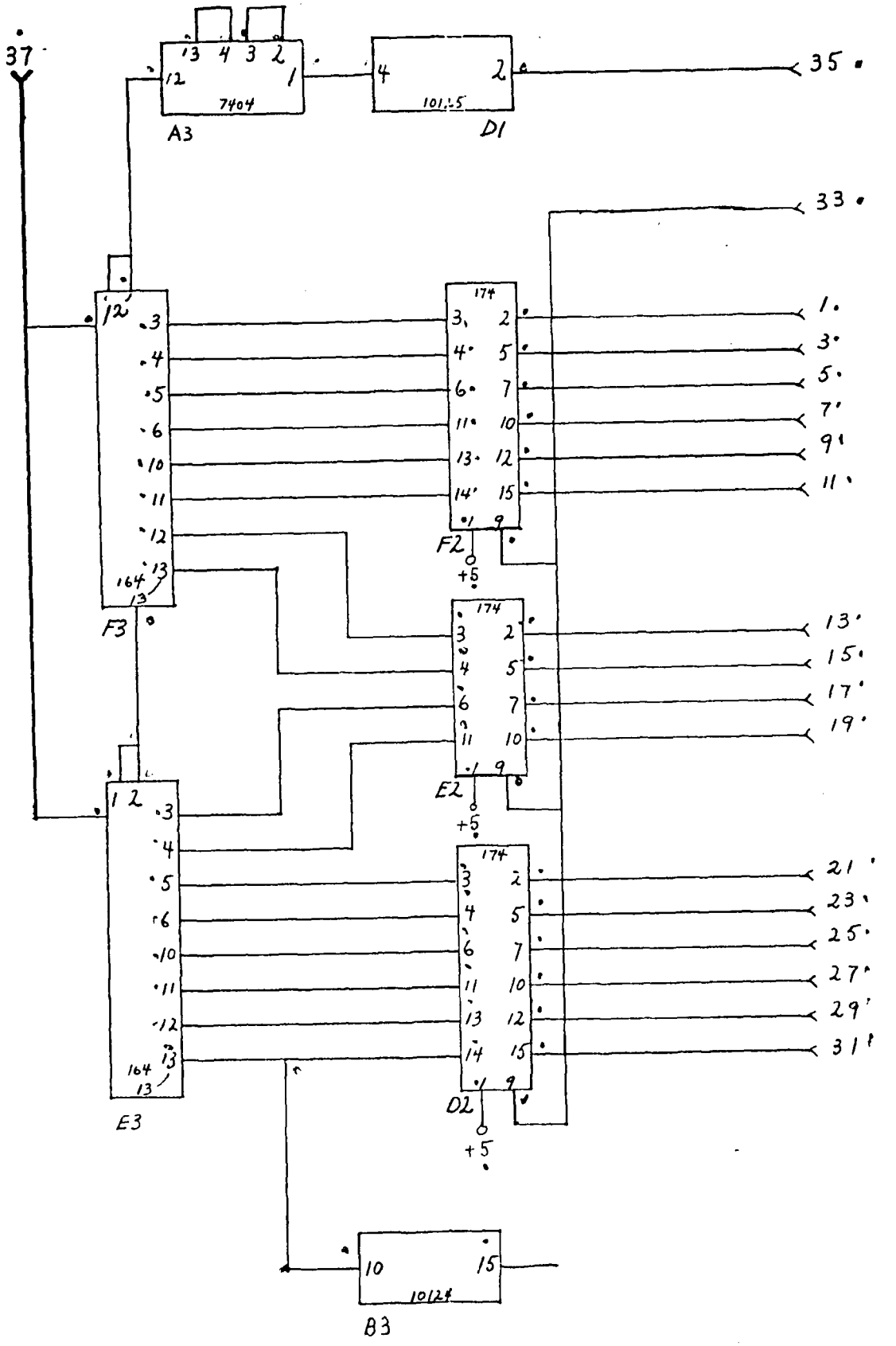
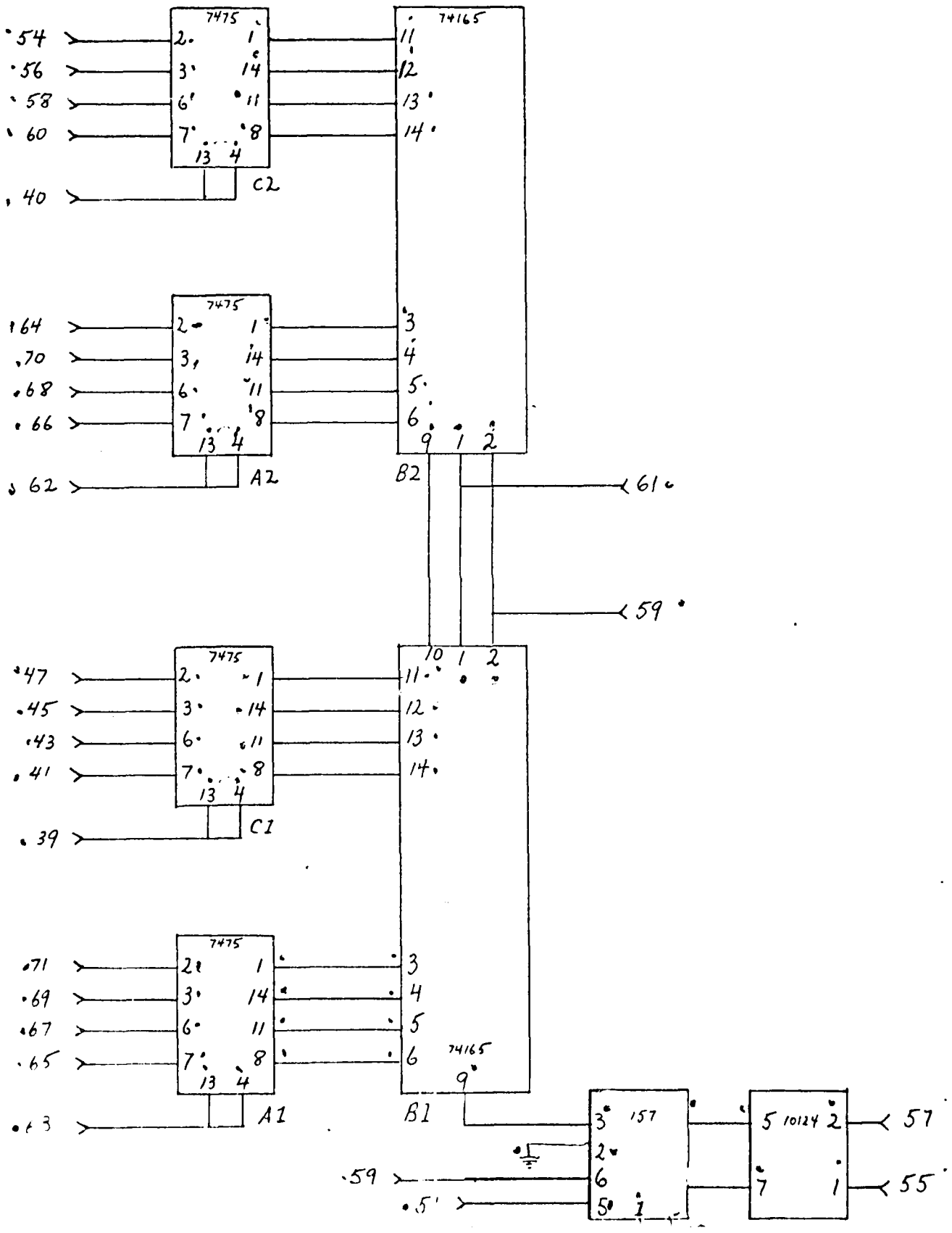


Fig. 2.1-3 Parallel to Serial Converter



## 2.2 Effect of Packet Destruction

Let us assume that a packet consists of 1000 information bits. Then, it can be shown that at a bit rate of 16 Mb/s each packet contains the bits for a complete line of video.

If channel noise produces an error, thereby destroying a packet, the result is the elimination of a line of video. However, the effect of the channel noise can be significantly reduced if error correction coding is employed since an error rate of $10^{-3}$ (which is quite large) implies a single bit error/packet. If error correction is not used, then the probability of each packet being in error will be quite high at the BER of $10^{-3}$. At a bit error rate of $10^{-5}$, a system with no error correction will have 10 to 5 lines in error/frame. As a result of the above observation we conclude that error correction is required in each packet if the error rate can reach $10^{-3}$.

Even with some error correction a random error burst may cause an occasional packet to be destroyed. When this occurs our studies have shown that the next packet should be written twice, once in the line position of the destroyed packet and once in the correct line location. Since the vertical resolution is somewhat greater than is actually needed this vertical smearing of a line is not noticeable at packet error rates of 1 error per 100 packets.

It is interesting to note that if we were encoding at the rate of 8 Mb/s, which is the lower limit for acceptable quality, then a destroyed packet means two lines have been eliminated. Fortunately, the two lines are <u>not adjacent</u> but are on the same field. Thus they are indeed separated by the correct line from the other field and the technique can still be employed.

The above "filtering" technique was preferred to the more classical techniques since it is readily implemented and requires <u>no</u> additional memory.

## 2.3 Frame-Change Detection

It is extremely simple to detect an initiation and completion of a voice signal. For, when there is no voice the ADM output is ...11001100... and at the onset of voice the first three to six bits are each "1" or "0", i.e. ...11001111... or ...1100110000.....

To electronically detect the presence of a frame change it is necessary to monitor the signals between frames. One obvious, albeit extremely complicated, way is to substract the pictures from two adjacent frames. If the magnitude of the difference signal exceeds a threshold we decide that the frame content has been sufficiently altered so as to require a new frame be transmitted. This technique could be used if we were employing an analog memory. However, since we are employing a digital memory this technique is not practical.

Another technique is to monitor one or more pixels in each frame. For example, consider monitoring the first pixel of the odd fields. Then an ADM which samples this pixel operates at the rate of 30 bits/s. As long as the output pattern is ...1100... the frame information has not changed. However, as soon as three -"1"'s or three -"0"'s are detected we know that the frame has changed.

Of course, one could monitor pixel (i.j) located somewhere in the center of the picture or could use several detectors. However, to verity our procedure we chose a single pixel.

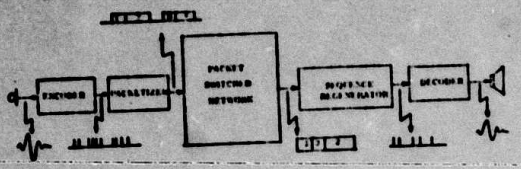



FIG. 1 PACKET VOICE TRANSMISSION SYSTEM

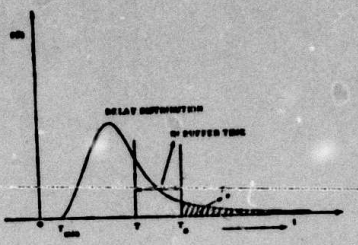



FIG. 3 DELAY DISTRIBUTION AND SEQUENCE REGENERATION

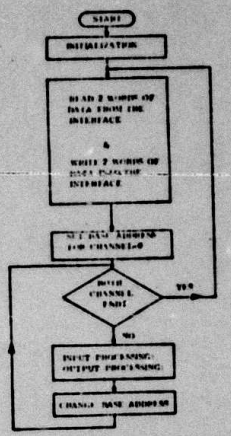



FIG. 5 GENERAL FLOW CHART OF THE SIMULATION

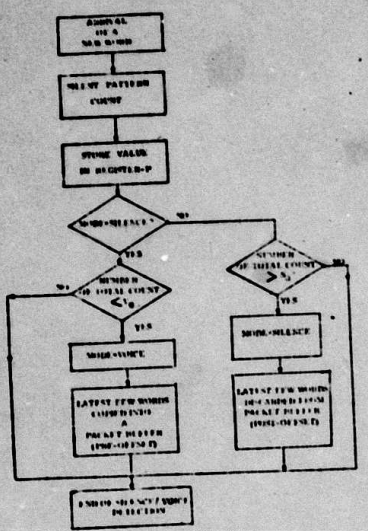


FIG. 2 SILENCE/VOICE DETECTION SCHEME

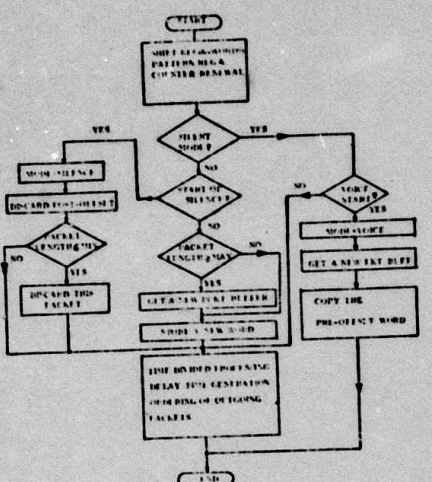
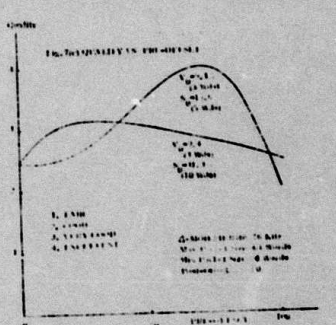



FIG. 6a FLOW CHART OF INPUT PROCESSING



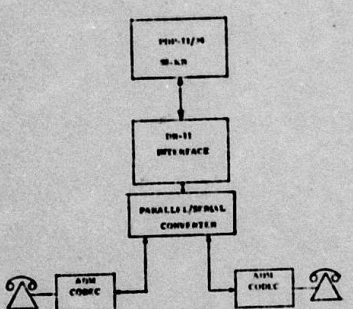



FIG. 4 PACKET VOICE NETWORK SIMULATION

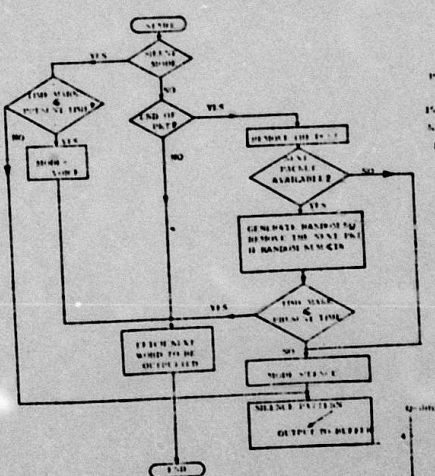
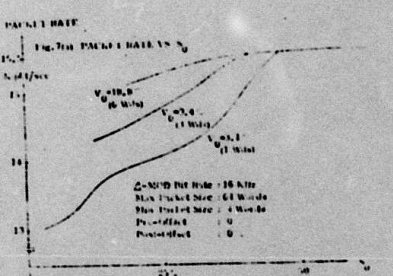
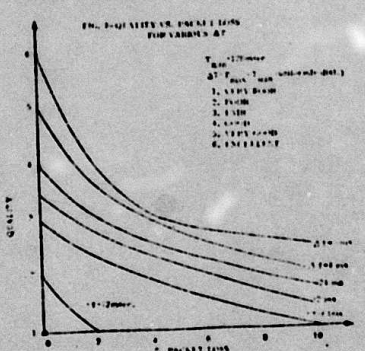
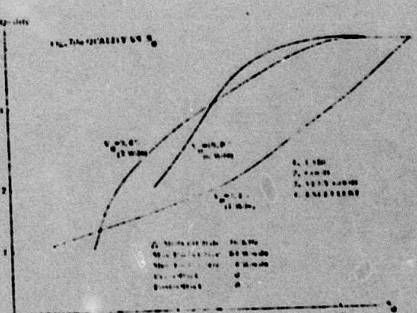



FIG. 6b FLOW CHART OF OUTPUT PROCESSING



| | | | | | |
|---|---|---|---|---|---|
| | 1 | 1.3 | 1.1 | 1.2 | 2.2 |

Table 1.