LEVEL III

See A103413 Pt B

ADA008563

ARPA QR6
PART A

AD A103412

IMAGE TRANSMISSION VIA SPREAD SPECTRUM TECHNIQUES.

Part A.

ARPA Order-2303

DTIC
ELECTE
AUG 2 7 1981
S D
H

Robert W. Means
Jeffrey M. Speiser
Edwin H. Wrench
Harper J. Whitehouse

ARPA Quarterly Technical Report. no. 6, 1 Jan-31 Dec 75,

January 1, 1975 – December 31, 1975

Advanced Research Projects Agency
Order Number 2303
Code Number 3G10

1 Jan 76

January 1, 1976
Naval Undersea Center
San Diego, California 92132

Approved for Public Release; Distribution Unlimited.

DTIC FILE COPY

390458

81 8 26 086

IMAGE TRANSMISSION
VIA
SPREAD SPECTRUM TECHNIQUES

Investigated by
DR. ROBERT W. MEANS (714:225-6872), Code 408
JEFFREY M. SPEISER (714:225-7621), Code 408
DR. EDWIN H. WRENCH (714-225-6871) Code 408
and
HARPER J. WHITEHOUSE (714:225-6315), Code 4002
Naval Undersea Center
San Diego, California

Sponsored by
Advanced Research Projects Agency
Order Number 2303
Code Number 3G10
Contract
effective: 15 February 1973
expiration: 30 June 1976
amount: $1,599,400

ARPA Quarterly Technical Report

January 1, 1975 – December 31, 1975

Form Approved Budget Bureau No. 22-R0293

DTIC
AUG 2 1981
H

PREFACE

This progress report appears in two parts. Part A is a summary of work done in support of this program at the Naval Undersea Center. Part B contains final reports submitted by contracts in support of this program.

The major milestone achieved during this period is the demonstration of the TV bandwidth reduction system CCD implementation. Still pictures and video tapes are produced at various bit rates. System specifications are documented and contracts and negotiations are in progress to have a flight system produced. This system is described in Appendix A.

Also accomplished are simulation studies on more advanced bandwidth reduction systems that take advantage of both spatial and temporal correlation in TV images. Results of this work are described in Appendices B and C.

The algorithm for the prime cosine transform is documented in Appendix D, and a surface wave prime Fourier transform device is implemented for evaluation of the algorithm in Appendix E.

Hardware developed under this program is shown to have applications in many areas of signal processing and techniques for modularly expanding the basic building blocks of this hardware into larger systems are investigated in Appendix F. Several signal processing architectures are described in Appendix G.

Simulation studies by USC and others document that the Cosine transform is very close to the Karhunen-Loève transform when applied to image data. These results are obtained from experimental data but as yet no analytical explanation is available. However, NUC has undertaken to give a theoretical basis for the performance of the cosine transform and the results are presented in Appendix H.

Papers have been presented at several conferences on the BBD and CCD bandwidth reduction systems and are included as Appendices I, and J.

Work is also conducted on a frequency hopping modem and some preliminary results are presented in Appendices K and L.

For ease of content location, the enclosed appendices are numbered consecutively in Arabic numerals beginning with the title page of Appendix A and continuing to the end of Appendix L.

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

CONTENTS

Preface . . . iii

APPENDICES

A. Hybrid Cosine/DPCM TV Bandwidth Reduction Hardware . . . 1

B. Combined Spatial and Temporal Coding of Digital Image Sequences . . . 49

C. Interframe Transform Coding and Predictive Coding Methods . . . 61

D. The Prime Cosine Transform . . . 69

E. Prime Transform SAW Device . . . 77

F. Modular Implementations of the Discrete Cosine Transform . . . 83

G. Signal Processing Architectures Using Transversal Filter Technology . . . 91

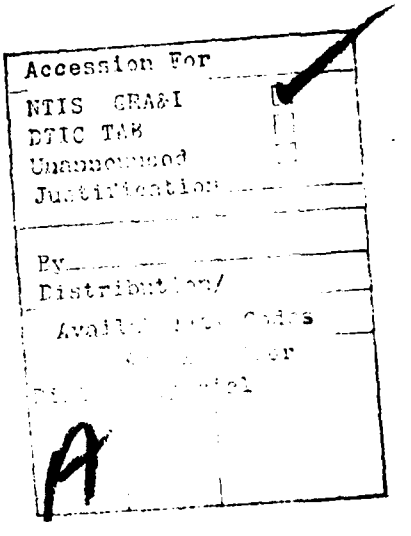
H. Investigating Adaptive Transforms in Information Processing Applications . . . 119

I. Real Time Television Image Bandwidth Reduction Using Charge Transfer Devices . . . 163

J. Real Time TV Image Redundancy Reduction Using Transform Techniques . . . 177

K. Frequency Synthesis via the Discrete Chirp and Prime Sequence ROMs . . . 185

L. Exponential Residue Codes . . . 209



APPENDIX A

HYBRID COSINE/DPCM
TV BANDWIDTH REDUCTION HARDWARE

HYBRID COSINE/DPCM TV BANDWIDTH REDUCTION HARDWARE

by Robert Means and E. H. Wrench Jr.

Following is a technical description of the bandwidth compression system developed at the Naval Undersea Center. This paper is an excerpt from the specifications for a contract currently under negotiation to produce flight hardware for the Army AQUILA RPV.

Section I. Bandwidth Reduction System Specifications

1.1 The item required is a system which can reduce the data required to transmit television images over a digital communication modem to the rates specified below. The system performance, in terms of picture quality and data rate, must approach that observed in computer simulations. The system must be capable of accepting EIA monochrome television signals.

1.2 Considerable simulation, analysis and breadboard experimentation have been done on the problem of optimizing the bandwidth compression that can be obtained by a real time system within a small power, weight, size and cost constraint. The system will be used in the Aquila mini remotely piloted vehicle. Figure 1.1 is a block diagram of the hardware required. It must accept composite video in the flight hardware (transmitter) and return composite video at the ground station (receiver). The data rates out of the source encoder are specified below. Contained at the end of this paper are schematics of the subsystem which have been implemented and tested at the Naval Undersea Center. These are provided for informational purposes. They should be considered as design guidelines which would aid the contractor in meeting system specifications.

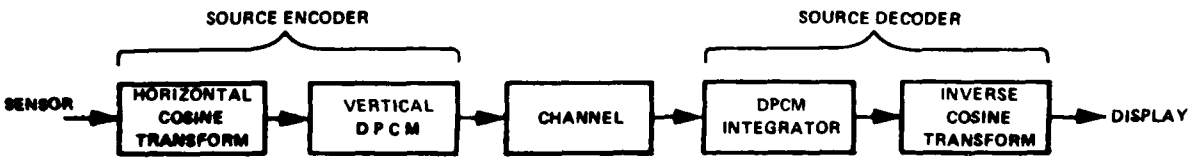


Figure 1.1

1.3 Each specification will be expanded upon in a separate section where appropriate. Some of these requirements are rigid and must be adhered to by the contractor. Other requirements are such that considerable design effort, invention and ingenuity must be expended by the contractor to meet the performance specifications. It is realized by the government that the required system imposes performance specifications that will require novel techniques to achieve and that there are tradeoffs between power, weight, cost and performance. Specifications will be found in each section. A partial list follows

1.4 Timing

1) The number of picture elements in each horizontal line must not be appreciably less than 256.

2) The number of lines per field must be 262.5.

3) The frame rate must be 7.5 frames per second with the capability of operating at a reduced frame rate with a minimum of system redesign. Some design

effort should be expended to insure that the system would not be completely incompatable with a solid state sensor operating in the snapshot mode.

4) The video signal must be transformed in horizontal blocks of either 31 or 32 pixels.

5) Video sample rate = 4.8 MHz.

6) System clock frequency = 9.6 MHz.

7) There must be a sync output pulse every 8 fields.

1.5 Discrete Cosine Transform

1) The horizontal transform must be a discrete cosine transform.

2) Dynamic range of output coefficients must be $\geq 2^8$.

3) Peak-to-peak signal to peak-to-peak noise of output coefficients must be $\geq 2^8$.

4) Transform coefficients must be accurate to within one percent over the whole dynamic range of the transform.

1.6 Differential Pulse Code Modulator

1) The vertical transform must be a differential pulse code modulator operating on cosine transform coefficients.

2) Quantizer must be within the DPCM feedback loop.

3) Dynamic range at input must be 2^8.

4) Maximum error at output with steady state input signal must not exceed one least significant bit when quantized at 6 bits per coefficient.

5) Quantization levels must be spaced for equal probability of occupancy at all bit rates.

6) DPCM must be capable of variable bit assignment per coefficient from zero to six bits.

7) Output rate must be continuous serial capable of switching between 200, 400, 800 and 1600 kilobits per second on command from the modem.

1.7 Ground Station

1) The ground station must be implemented with digital hardware.

2) Power and weight are not critical factors in the ground station.

1.8 Form Factors

1) The system must fit on one (1) card, the size of which is approximately 4" x 8" x 1/2". Exact form factors will be specified after award of contract.

2) The system must have a weight of less than one pound.

3) The system must dissipate less then ten watts.

4) Environemntal specifications are those specified for the Aquila Lockheed RPV.

1.9 Administrative

1) There will be design review meetings held every 90 days at the contractors plant.

2) Production cost estimates of the system are requested with a production run of 1000 a year for 5 years to be used as a basis.

3) A bid is requested on a package consisting of two airborne subsystems and one ground station.

4) A bid is requested on a package consisting of 3 airborne subsystems and two ground stations.

5) Proposal evaluation criteria are listed in order of importance.

   a) weight, size, power
   b) performance and design
   c) production cost
   d) contractor experience and personnel
   e) risks inherent in design
   f) bid prices

6) Delivery date is 9-10 months after award of contract.

Section II. Timing and Synchronization

2.1 The TV bandwidth reduction system must operate synchronously with the TV camera and the Harris modem. The clocks required to insure compatibility have been determined and will be supplied by Harris. These clocks will be available to operate both in the plane and on the ground at synchronize time and phase relationships.

2.2 The TV sync in the plane will be supplied by the camera. On the ground the TV sync must be reconstructed by the bandwidth reduction system from a timing pulse transmitted from the plane and the system clocks. The detailed operation of the system timing is described below.

2.3 In the plane, the TV sync is generated by a TV sync generator chip that is driven by a 2.0 MHz clock. The 2.0 MHz clock is supplied by Harris and is derived from the system clock. (2 MHz = 4.8 MHz ÷ 12 x 5.) Each TV line is composed of 130 of the 2 MHz clocks for a line time of 65.0 μs. The V and H signals generated by the sync chip are supplied to the bandwidth reduction system.

2.4 The basic sample clock rate for the bandwidth reduction system is 4.8 MHz and is derived from a 9.6 MHz clock supplied by Harris. The TV line time corresponds to 312 clocks at 4.8 MHz. Of these, 256 correspond to the active TV line, and 56 to retrace time. The even field is processed as though it were identical to the odd field, so the vertical resolution is 262.5 lines.

2.5 The phase relation of the horizontal sync pulse to the 4.8 MHz clock is not known but will remain constant since the horizontal sync is derived from the 2 MHz clock which in turn is derived from the 4.8 MHz clock. Therefore each TV line will start at the same time relative to the 4.8 MHz sample clock. This permits the 4.8 MHz clock to be used to partition the active TV lines into 256 pixel blocks. The pixels are numbered with 0 on the left and 255 on the right.

2.6 The effective frame rate of the TV is reduced by a factor of eight by only coding 1/8 of a TV line during each horizontal time interval (65 μs). Eight TV fields must be swept out by the camera before one entire field has been coded. During the first field, pixels 0-31 of all lines will be coded and transmitted. During the second field pixels 32-63 will be coded, etc.

2.7 The blocks of 32 pixels form vertical bars in the TV field are referred to as stripes. These stripes are numbered 0-7 with stripe zero containing pixels 0-31. A stripe counter is incremented by the vertical sync pulse obtained from the camera.

2.8 A pulse that exactly describes the beginning of stripe zero (frame sync) is used to obtain video synchronization between the plane and the ground. The frame sync pulse (7.5 Hz rate) is sent to the status modem in the plane and then transmitted to the ground. At the ground the pulse is obtained from the status modem with its time relative to the data preserved. The frame sync is then used to synchronize the pixel counter, line counter, and stripe counter on the ground with those in the plane. It is also used to lock a TV sync chip (required to produce composite video) on the ground with the one in the camera. Therefore, complete synchronization between air and ground video is maintained by the frame sync signal.

2.9 Another area where synchronization is required is the interface between the TV bandwidth compression system and the Harris modem. The interface requires that serial data be passed from the TV system to Harris at four fixed rates: 200, 400, 800, and 1600 K bits/sec. This corresponds to, respectively, 13, 26, 52, and 104 bits/TV line. The output bits are to be synchronous with a data clock supplied by Harris at the bit rate. The phase of this clock relative to the 4.8 MHz clock is not known but will remain fixed at each of the four frequencies.

2.10 The interface at the ground between the modem and the TV system is identical to the one just described. The data from the modem is presented serially and synchronously with the data clock from the modem. The data rate at the interface is determined by the state of 2 bits (x&y) supplied to the system by the command and control modem.

Section III. Discrete Cosine Transform

3.1 There are two discrete cosine transforms which can be defined for a finite block size data set. They have been called the odd discrete cosine transform (ODCT) and the even discrete cosine transform (EDCT). For a complete discussion of the differences see Appendix A.

3.2 There exists many possible algorithms to implement the discrete cosine transform. It is not the purpose of this procurement to specify a given algorithm. However, the hardware implementations of each approach considered by the Naval Undersea Center will be discussed.

Chirp Z Algorithm

3.3 Consider the odd cosine transform. The chirp Z algorithm computes the coefficients of a 32 point data block by implementing the equation

$$G_k = e^{-i\pi k^2/63} \sum_{n=0}^{32} e^{-i\pi n^2/63}\, e^{+i\pi(k-n)^2/63}\, \hat{g}_k$$

where

$$\hat{g}_k = \begin{cases} g_o/2 & k = 0 \\ g_k & k = 1 \ldots 31 \end{cases}$$

A block diagram of the hardware is shown in figure 3.1. This algorithm was implemented by the Naval Undersea Center with charge coupled device transversal filters developed by Texas Instruments. It and the rest of the bandwidth compression system has been demonstrated with real time video. The complete schematics for the system are attached at the end of this paper.

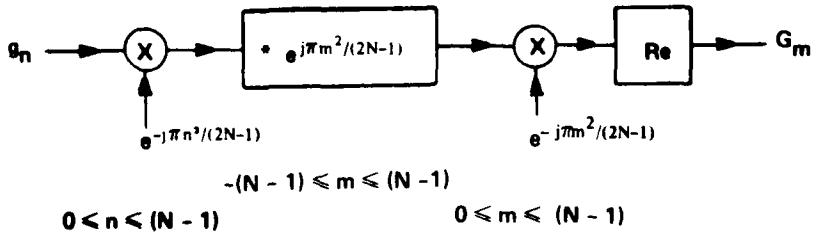


Figure 3.1. Block Diagram of Chip Z Algorithm Hardware.

3.4 The hardware consists of three major blocks: The premultipliers, the transversal filters, and the postmultipliers. All three blocks performed marginally at the 4.8 MHz video sample rate. The system performance was close to, but did not equal, that of the computer simulations.

3.5 The breadboard system built by NUC used an LSI chip developed by Texas Instruments. All the premultipliers and differential current integraters were placed on the device. The transversal filters themselves performed adequately at the sample rate of interest (4.8 MHz), but the peripheral devices - multipliers and differential current integrators - did not. In the breadboarded system these were implemented by separate modules or discrete components. It was the off chip modules which determined the size of the laboratory breadboarded system. In order to meet the contract specifications it is necessary to put as many of these peripheral devices on the chip as possible; i.e. a new CCD chip is required.

3.6 Since the processor has a complete line to process 32 samples, it is possible to capture the video in an analog delay line or memory at high sample rate (4.8 MHz) and read it out into the processor at a slow rate (1.2 MHz). This option would relax the speed requirements on the multipliers, differential current amplifiers, and CCD's. It will also relax the requirements on the DPCM. It could, however, add a separate piece of hardware to the system.

3.7 The premultiplier should have an accuracy of at least 6 bits. It is presumed that the video is limited to 6 bit accuracy, probably in the display. The transform coefficients should have a dynamic range of at least eight bits.

Prime transform algorithm

3.8 The prime transform algorithm can be implemented by the hardware shown in figure 2. The prime transform requires the use of a permuting analog memory, a transversal filter and a second permuting memory at the output. In addition, it requires a separate computation of the D.C. coefficient. Also the first video sample must be treated different than all the rest. An analog memory chip has been developed by the Reticon corporation under a contract by the Naval Undersea Center. This system has not been breadboarded by the Naval Undersea Center and the memory chip has not been sufficiently tested. However the performance specifications claimed by Reticon are attached in the appendices. The prime algorithm for a Fourier transform, is

$$G_R k' = g_o + \sum_{n'=1}^{N-1} g_R n' \, e^{-i2\pi R^{k'+n'} N}$$

where

$$k = R^{k'} \quad (\text{Mod } N)$$

$$n = R^{n'} \quad (\text{Mod } N)$$

and R is a primitive root of N.

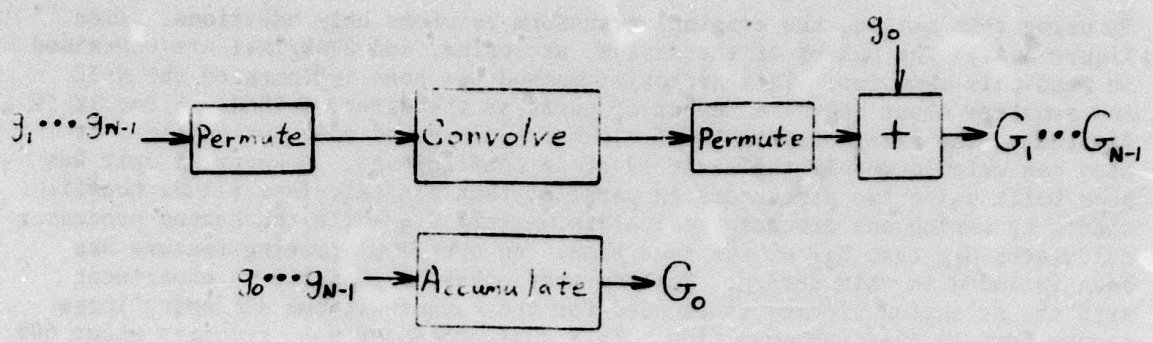


Figure 3.2. Prime Transform Algorithm.

Direct Computation

3.9 The calculation of the discrete cosine transform could be done by straightforward digital techniques. Two methods of performing this digital cosine transform have been investigated. The first method is a direct implementation of the algorithm while the second involves a trigonometric substitution to reduce the complexity of the calculation.

3.10. The odd discrete cosine transform of the form

$$G_k = g_o + 2 \sum_{n=1}^{N-1} g_n \cos \frac{2\pi nk}{2N-1} \quad k = 0,1,2,\ldots N-1$$

has been directly implemented for N=32. The design uses a read-only-memory to store the basis vectors and performs the multiplication using LSI multipliers. (See Figure 3.3.) Due to the limitation of a clock frequency of 9.6 MHz and a maximum of 65 μsec in which to perform the calculation, only the first 19 coefficients can be calculated. In order to obtain all 32 coefficients, it is necessary to have at least 2 parallel processors operating together.

3.11 The transform can be implemented without having to do any multiplies by the substitution of variable $F_n = \cos^{-1} g_n$. The transform then becomes

$$G_k = \cos f_o + 2 \sum_{n=1}^{N-1} \left[\cos \left(\frac{2n\pi k}{2N-1} + F_n\right) + \cos \left(\frac{2\pi nk}{2N-1} - F_n\right)\right] \quad k = 0,1\ldots N-1$$

By using this method, the original transform requires only additions. (See Figure 3.4.) The values of the cosine, arccosine, and $2\pi nk/2N-1$ are contained in read-only-memories. This arccosine method has been implemented for N=32 and requires about the same number of parts as the direct method. Since it is limited by the same constraints of clock frequency and calculation time, it also can calculate only the first 19 of 32 coefficients. However, a unit has been built using two processors in parallel that can calculate all 32 coefficients by having one processor calculate G_0 thru G_{15} while the second processor calculates G_{16} thru G_{31} at the same time. An overflow limiting feature has been included in this design. This feature makes it possible to experiment with the selection of bits to be used for the output without suffering large errors from an overflow condition. This dual processor unit requires about 60% more parts than the single unit. Both the direct implementation and the arccosine method units use double buffering at both the input and output so that the processor can be operating on data while other data is read in and results read out. This increases the time available for calculation. Schematics for these systems are attached at the end of this paper.

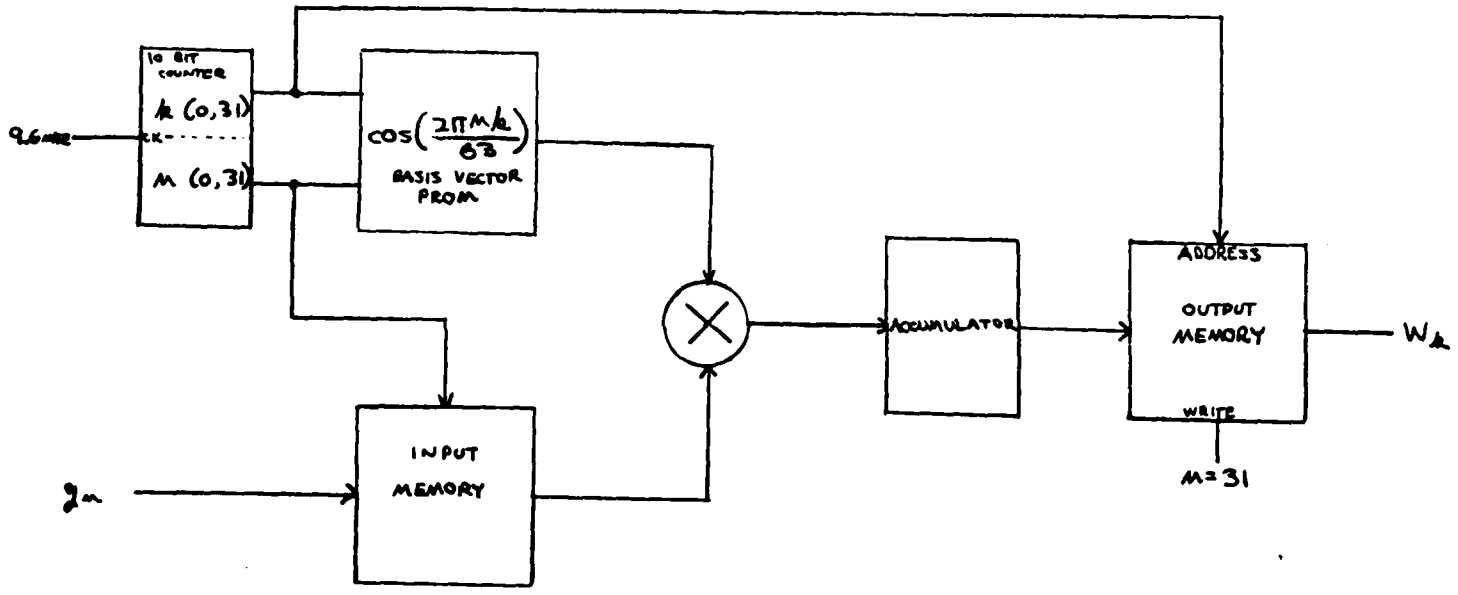


Figure 3.3. Direct Implementation of DCT Block Diagram.

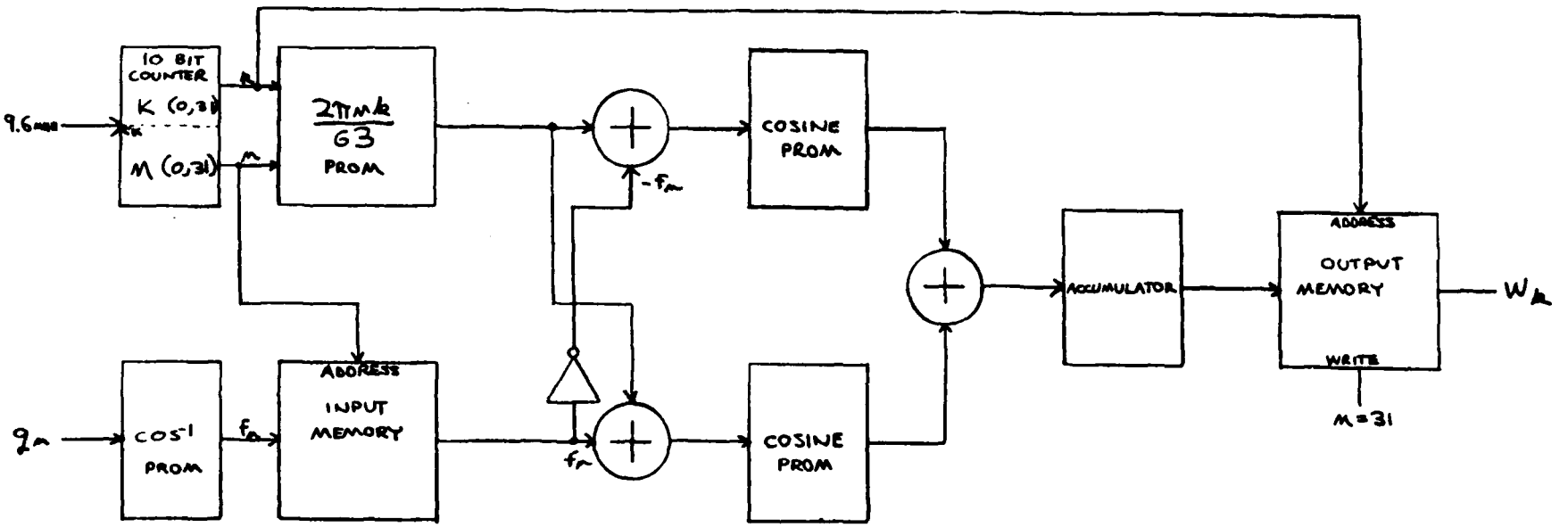


Figure 3.4. Arc-Cosine Implementation of DCT Block Diagram.

Section IV. Prewhitening Filter

4.1 There are two reasons to use a prewhitening filter as the first operation on the video signal.

4.2 One: Since the cosine transform compacts the sampled video into the low order transform coefficients, and the subsystem which calculates the cosine transform has a limited dynamic range, the high frequency coefficients will have a poor signal to noise ratio. One method of increasing the signal to noise ratio for the high frequency coefficients is to filter the video to emphasise the high frequency information. The phase requirements and invertability of this filter and their implications will be discussed later.

4.3 Two: The DPCM requires a non-linear quantizer. The choice of the quantization levels is extremely important to the resulting picture quality. The levels are chosen by examining the probability distribution for the difference between transform coefficients. Since the variance of the transform coefficients decreases as a function of n(coefficient number), each coefficient requires a separate quantizer. This requirement can be eliminated by normalizing all the coefficients to have the same variance. The normalization of the coefficients can be done by multiplying each coefficient by the appropriate scale factor as it enters the DPCM subsystem. However, the high frequency coefficients would still have a poor signal to noise ratio. Instead, since this operation on the transform coefficients is equivalent to a frequency domain multiply, one can implement the scaling necessary for the DPCM by a prewhitening filter directly on the video.

Design considerations of the prewhitening filter

4.4 Let us assume that the variance of the cosine transform coefficients decreases proportional to 2/(n+2). (This is a fair approximation for most TV scenes.) The "ideal" prewhitening filter would multiply the coefficients by (n+2)/2. This filter is very difficult to implement in the time domain. However, one can implement a simple filter which approximates the "ideal" function. The difference between the approximation and the "ideal" equalization can be removed by the use of a multiplier before the DPCM. This multiplier then operates on coefficients with good signal to noise ratio and only has to multiply coefficients by a relatively small factor. A simple example of a prefilter which can be implemented by a delay line and a differential amplifier is shown in figure 4.1

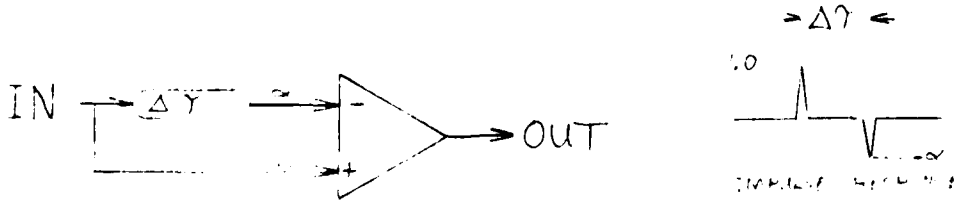


Figure 4.1. Simple Example of a Prefilter.

The impulse response of this filter is also shown. In the frequency domain this filter multiplies the Fourier transform coefficients by the transfer function

$$H(f) = 1 - \alpha + 2i \sin \pi f \Delta\tau$$

4.5 There are severe limitations on the filter phase distortion allowed. The filter in the example has complex phase shift. It is not possible to compute a cosine transform of the video and use this prefilter. If α=1.0, then it is possible to compensate for the purely imaginary phase shift, but it introduces problems with dc signals. The filter should multiply the variance of the cosine transform coefficients by the transfer function of the filter. Since the cosine transform of the video is the Fourier transform of the extended data set (Appendix A) one can define:

$g(t) \equiv$ sampled video data set

$g_e(t) \equiv$ extended data set $\equiv g(t) + g(-t)$

$h(t) \equiv$ impulse response of the prefilter

$F(g(t)) \equiv$ Fourier transform of $g(t) \equiv G(f)$

$C(g(t)) \equiv$ cosine transform of $g(t) \equiv F(g(t) + g(-t)) = G(f) + G^*(f)$

Thus

$$C(h(t) * g(t)) = F(h(t) * g(t) + h(-t) * g(-t)) \qquad (1)$$

$$= H(f)\ G\ (f) + H^*\ (f)\ G^*\ (f)$$

$$= \text{Re}\ H(f)\ G\ (f) = \text{Re}\ F(h * g)$$

If H(f) is purely real, then the cosine transform of h * g can be expressed as

$$C(h * g) = H(f)\ [G + G^*] \qquad (2)$$

Thus we have multiplied the cosine transform coefficients by the transfer function of the prefilter. It appears that a prefilter with a real transfer function is the simplest one to achieve the proper whitening. Such a filter is slightly more complicated to build. The impulse response of such a filter is given in figure 4.2

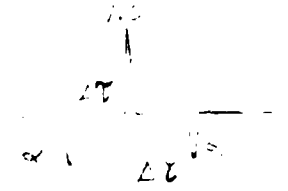


Figure 4.2. Filter Impulse Response.

In the frequency domain this filter has the transfer function

$$H(f) = 1-2\alpha \cos 2\pi f\Delta\tau$$

4.6 It has been assumed that the inverse filter will be implemented digitally in the ground station. The inverse of the filter shown in figure 4 cannot be implemented easily. However, this is at the ground station. The impulse response is shown in figure 4.3

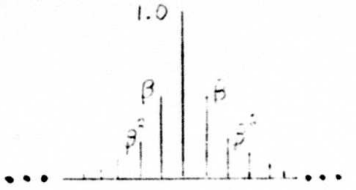


Figure 4.3

where

$$\beta = \frac{1}{2\alpha}\left(1-\sqrt{1-4\alpha^2}\right) \quad \text{for } \alpha < 0.5$$

This inverse can be approximately implemented by the hardware shown in figure 4.4

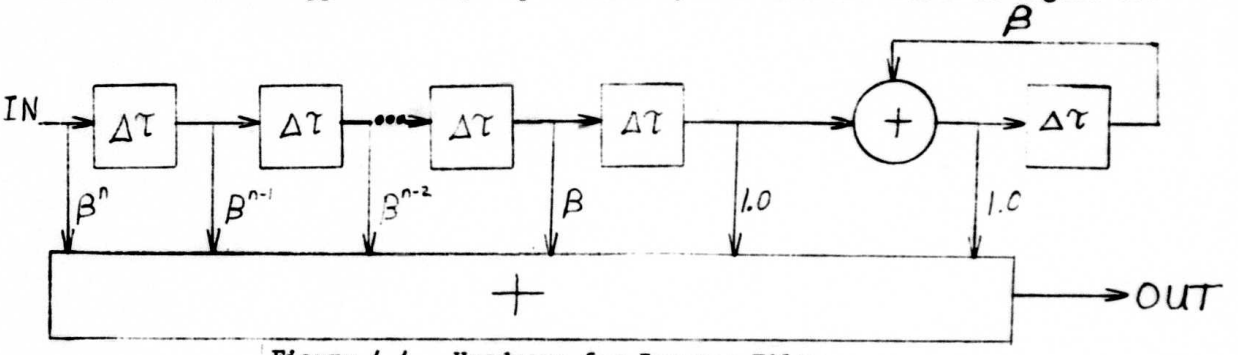


Figure 4.4. Hardware for Inverse Filter.

4.7 The accuracy of this filter will depend on the number of stages in the nonrecursive part and the value of β. For example: for a pre-emphasis filter with α=0.4, then β=0.5 and with eight stages in the non-recursive filter, the accuracy of the output will be greater than one part in 2^8. Output is valid after n+1 stages.

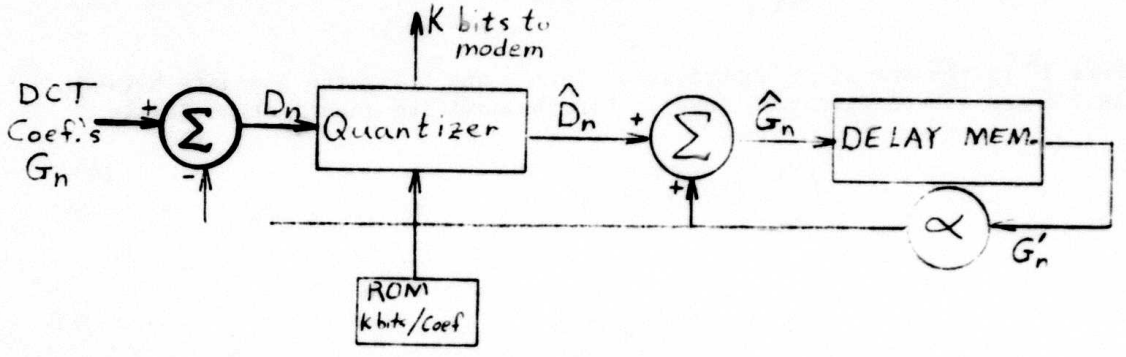
4.8 It appears to be preferable to implement the prefilter which has a real transfer function for several reasons: the cosine transform does not have to be modified; noise does not accumulate in the inverse filter; and, most important, a dc bias on the video signal reaches the cosine transform as dc.

4.9 The discussion of the prefilter in this section has gone into detail on a number of alternative designs. Attention was paid to the impact on the algorithms. It was not meant to give the impression that the prefilter design is extremely critical to picture quality. Several different prefilters have been implemented by NUC and the differences in picture quality were observed to be small. It was important, however, to implement some type of pre-emphasis and de-emphasis. The pre-emphasis should process the whole line of video so that the de-emphasis filter operates across the boundaries of the horizontal interlace used in the system.

Section V. Differential Pulse Code Modulator

5.1 The differential pulse code modulator (DPCM) is the system element that produces the actual bandwidth reduction. The processes of prewhitening and cosine transfering are exact and invertable and therefore no bandwidth reduction occurs. The DPCM however is not exactly invertable because approximations are made in the quantization of the differences between coefficients. The use of the approximate difference results in a reduction in the amount of data required to transmit the coefficients. The transmitted data rate is varied by changing the number of quantization levels used for each coefficient.

5.2 The basic structure of the DPCM is as shown below.



The operation is as follows. The DCT coefficients (G_n's) enter in time sequence, and are differenced with a fraction (α) of the corresponding coefficient from the previous line (G'_n's). The difference $\hat{D}_n$ is then approximated by one of 2^k possible values by the quantizer. The value of k for each coefficient is programmed in a ROM. The k bits from the quantizer that define the quantization level are then transmitted serially to the modem. The quantized difference, $\hat{D}_n$, is then added to $\alpha \cdot G'_n$ to produce an approximation, $\hat{G}_n$, of the input coefficient G_n. This value is stored in the delay memory to be used as G'_n in the next line.

5.3 It is important to note that by placing the quantizer inside the feedback loop the quantization error is kept from accumulating, and the error converges to the smallest quantization level in steady state.

5.4 The most critical portion of the DPCM is the design of the quantizer. For the coding of the coefficient differences to be efficient, all quantization levels must be equally likely. It has been shown by Habibi (Appendix J references)

and others that for a wide variety of pictures, the amplitudes of the coefficient differences are approximately exponentially distributed. The probability distribution for the differences is therefore assumed to be

$$f(D_n) = \alpha_n \cdot e^{-\alpha_n |D_n|}$$

where α_n is the variance of the nth coefficient. Since the coefficients have been prewhitened, all α_n's are approximately equal. The distribution is symmetric about zero, so only the positive quantization levels need be calculated. The probabilities of a difference falling into any quantization level must be equal, i.e.

$$\int_{b_i}^{b_{i+1}} f(x)\, dx = \frac{1}{2^k}$$

where 2^k is the number of quantization levels and b_i is the boundary between the i-1 and ith quantization level. For the positive quantization levels

$$\int_{0}^{b_i} f(x)\, dx = \frac{i}{2^k}$$

or

$$b_i = \frac{1}{\alpha} \ln \left[\frac{2^{k-1}}{2^{k-1}-i}\right] \qquad (A)$$

There are also boundaries at zero and at negative b_i.

5.5 Once the quantizer boundaries are determined, the quantization values, Q_i, are chosen to minimize the mean square quantization error, i.e.,

$$\overline{\varepsilon^2} = \overline{(x_i - Q_i)^2} = \text{Minimum where } x_i \text{ is an input to the quantizer that falls between } b_{i-1} \text{ and } b_i.$$

The value of Q_i is determined by setting

$$\frac{d\overline{\varepsilon^2}}{dQ_i} = 0 = \overline{2(x_i - Q_i)} = 2(\bar{x}_i - Q_i)$$

or $\bar{x}_i = Q_i$ where $\bar{x}_i$ is the mean of x_i for x_i contained in between b_{i-1} and b_i in quantization level i.

$$Q_i = \frac{\int_{b_{i-1}}^{b_i} xf(x)dx}{\int_{b_{i-1}}^{b_i} f(x)\,dx} \qquad (B)$$

5.6 The DPCM should have an input dynamic range of $\pm 2^7$. This implies that the difference between coefficients has a dynamic range of $\pm 2^8$. The input gain of the DPCM should be set such that the probability of the magnitude of the coefficient difference exceeding 2^8 is $1/2^p$ where $p=k_{max}$ (the largest number of bits that can be used in the quantizer) i.e.

$$\int_{-\infty}^{-2^8} f(x)dx + \int_{2^8}^{\infty} f(x)dx = \frac{1}{2^p}$$

or $$\int_{2^8}^{\infty} f(x)dx = \frac{1}{2^{p+1}}$$

so $$e^{-\alpha \cdot 2^8} = \frac{1}{2^p}$$

$$\alpha = 2^{-8} \cdot \ln 2^{p+1}$$

For the DPCM required in this specification, p=6. Thus α=0.0162. The quantization boundaries and mean values can be determined from equation A and B. The values are given in Table 1. It should be noted that to determine between which quantization level a difference falls, for k bits of quantization (k<6), it is necessary only to truncate the six bit quantization number to k bits. This will become clear by examining Table 1.

QUANTIZER POSITIVE VALUES FOR 64 LEVELS

QUANTIZER BOUND	QUANTIZER VALUE	TRANSMITTED BITS
0		
	1	000000
2		
	3	000001
4		
	5	000010
6		
	7	000011
8		
	9	000100
10		
	11	000101
13		
	14	000110
15		
	16	000111
18		
	19	001000
20		
	21	001001
23		
	24	001010
26		
	27	001011
29		
	30	001100
32		
	33	001101
35		
	37	001110
39		
	41	001111
43		
	45	010000
47		
	49	010001
51		
	53	010010
55		
	57	010011
60		
	63	010100
66		
	69	010101
72		
	75	010110
78		
	81	010111
85		
	89	011000
94		
	98	011001
103		
	108	011010
114		
	121	011011
128		
	137	011100
146		
	158	011101
171		
	190	011110
213		
	275	011111

TABLE 1. DPCM QUANTIZER VALUES

QUANTIZER POSITIVE VALUES FOR 32 LEVELS

QUANTIZER BOUND	QUANTIZER VALUE	TRANSMITTED BITS
0		
	2	00000
4		
	6	00001
8		
	10	00010
13		
	15	00011
18		
	20	00100
23		
	26	00101
29		
	32	00110
35		
	39	00111
43		
	47	01000
51		
	55	01001
60		
	66	01010
72		
	78	01011
85		
	94	01100
103		
	115	01101
128		
	147	01110
171		
	233	01111

TABLE 1. DPCM QUANTIZER VALUES (Cont'd)

QUANTIZER POSITIVE VALUES FOR 16 LEVELS

QUANTIZER BOUND	QUANTIZER VALUE	TRANSMITTED BITS
0		
	4	0000
8		
	13	0001
18		
	23	0010
29		
	36	0011
43		
	51	0100
60		
	72	0101
85		
	104	0110
128		
	190	0111

QUANTIZER POSITIVE VALUES FOR 8 LEVELS

QUANTIZER BOUND	QUANTIZER VALUE	TRANSMITTED BITS
0		
	9	000
18		
	30	001
43		
	62	010
85		
	147	011

TABLE 1. DPCM QUANTIZER VALUES (Cont'd).

QUANTIZER POSITIVE VALUES FOR 4 LEVELS

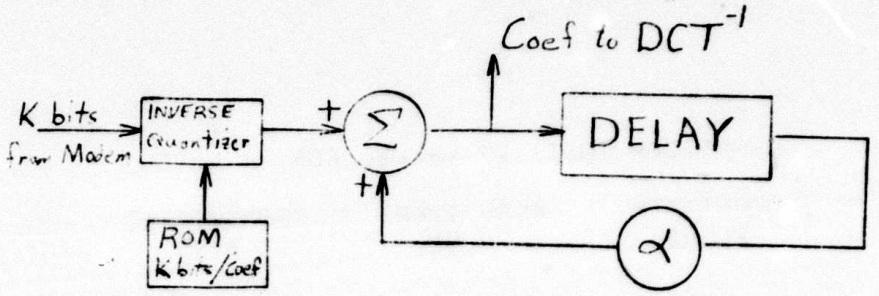
QUANTIZER BOUND	QUANTIZER VALUE	TRANSMITTED BITS
0		
	19	00
43		
	105	01

QUANTIZER POSITIVE VALUES FOR 2 LEVELS

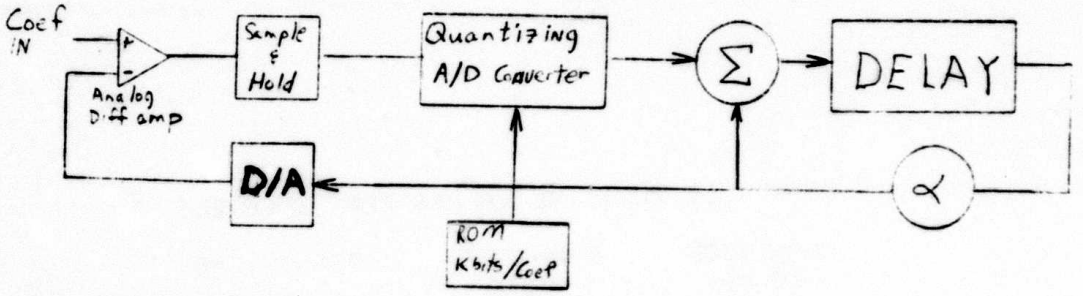
QUANTIZER BOUND	QUANTIZER VALUE	TRANSMITTED BITS
0		
	62	0

TABLE 1. DPCM QUANTIZER VALUES (Cont'd)

5.7 The inverse DPCM at the ground station is almost identical to the DPCM, only less hardware is required. The inverse DPCM is shown below.



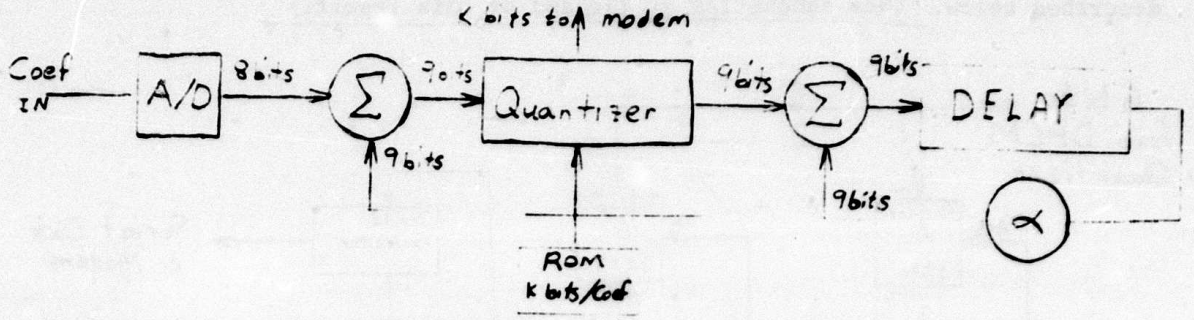
5.8 Two different designs for the DPCM have been built at NUC. The first is a hybrid analog, digital system shown schematically below



This design has the advantage that the analog to digital converter is nonlinear. It has 8 bits of dynamic range but only 4 bits of resolution. At low bit rates (i.e. large bandwidth reduction) 4 bits or 16 quantization levels are all that are required. Therefore, a very fast A/D and DPCM can be implemented using 16 comparators. This DPCM design has two disadvantages. 1) The analog differential amplifier and D/A converter at the input can introduce noise into the system. 2) For high data rates (low compression) it is desriable to have more than 16 quantization levels. At 1.6 m bits/sec, 6 bits or 64 levels are desirable for the first few low frequency coefficients.

5.9 A second all digital DPCM has been designed and is currently being implemented. It is intended to be used with a cosine transform that reduces the processing speed from 4.8 MHz to 1.2 MHz. A small relatively low speed 8 bit A/D converter can then be used directly on the output of the DCT. All DPCM arithmetic operations are then performed digitally. This approach uses somewhat less hardware since 1) it is running slower and 2) only one A/D is required rather than both an A/D and D/A. The disadvantage of the design is that the coefficients must enter the DPCM at a low rate (1.2 MHz) or a 4.8 MHz A/D must be used.

5.10 The block diagram of the all digital DPCM is shown below.

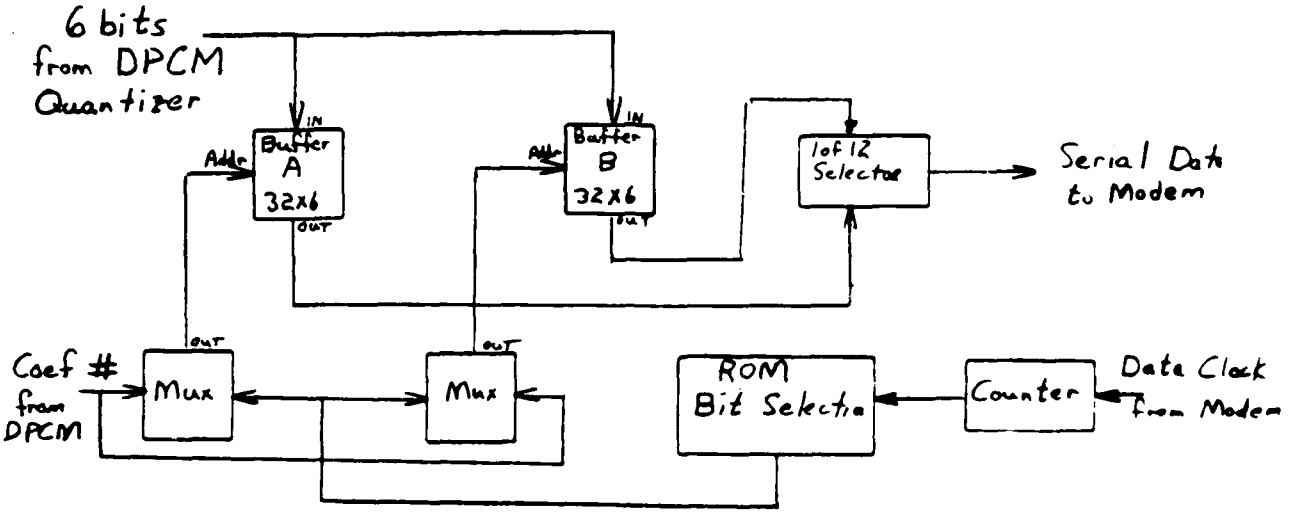


In both designs the quantization occurs in two steps. 1) The input data is assigned to one of $2^{k_{max}}$ quantization levels. 2) The k_{max} bits are then truncated to k bits as dictated by the ROM that stores the number of bits for each coefficient. These k bits and the bits for the number of bits per coefficient address a ROM that outputs the appropriate quantization value. The data for the ROM's for a 6 bit quantizer are given in Table 1. The input data rate for the DPCM is determined by the DCT. It is desirable that the DCT have an output rate less than the 4.8 MHz to reduce the speed at which the DPCM must operate. The output data rate for the DPCM is fixed by the modem interface requirements. The DPCM output is serial at four fixed rates: 200, 400, 800 and 1600 MHz. The required data rates translate to, respectively, 13, 26, 52, and 104 bits per TV line. The appropriate data rate is obtained by choosing the number of bits used to quantize each coefficient such that the sum equals the required number of bits per TV line. The optimum bit assignments for minimizing mean square error have been determined by Habibi at U.S.C. and are given in the table below. These bit assignment are not necessarily best, and the DPCM design should allow bit assignment changes to be made easily (i.e. changing a ROM). The DPCM must be able to switch between the four bit rates on command from the ground. Two input bits, x and y, are supplied by the command and control link that specify the bit rate, as shown below.

x	y	Bit Rate (kilobits/sec)
0	0	1600
0	1	800
1	0	400
1	1	200

5.11 The output data from the DPCM is serial and the bits are presented to the modem syncronously with a data clock at the appropriate rate (200 KHz, 400 KHz, etc.). The data clock is supplied by the modem. The data is generated by the DPCM at a nonuniform rate and a small buffer is required between the DPCM and the modem.

5.12 The interface from the DPCM to the modem is currently implemented and is described below. (See schematics at the end of this report.)



5.13 As mentioned earlier, when fewer than 6 bits are used to quantize a coefficient, only the high order bits of the 6 bit word are retained. The interface is a double buffered arrangement where all thirty-two, 6 bit quantization levels are stored. Only selected bits are read from the memory for transmission to the modem. A read only memory is addressed sequentially by the data clock from the modem. A ROM then reads the appropriate bits from one buffer, while new data is stored in the other buffer. At the end of each TV line the rolls of the buffers are interchanged. The bits that should be truncated because fewer than 6 bits are used for a coefficient are not addressed by the ROM and therefore not transmitted. The order in which the ROM addresses the bits is such that all the bits that must be sent at the 200 KHz data rate are transmitted first. Next, all bits required for the 400 KHz data rate (but not for the 200 KHz rate) are transmitted. This pattern continues until all bits have been accounted for. Since the ROM address is incremented by the data clock from the modem and reset to zero at the beginning of each TV line, the number of bits sent is determined only by the data clock frequency, and the correct bits are always transmitted.

5.14 The interface at the ground station is the reverse of the above process. A ROM stores the buffer address in which each input bit is to be stored. The quantized data is then reconstructed in one buffer while the other buffer feeds the inverse DPCM. Again, the role of the two buffers is changed at the beginning of each TV line.

5.15 This represents only one of many possible interfaces that could be designed and is probably not a minimum hardware implementation. An alternative approach is to use a first-in-first-out memory (FIFO) to buffer the data between the DPCM and the modem. This requires an extremely fast FIFO but could significantly reduce parts count.

5.16 An all digital DPCM is the more desirable implementation because it allows for 6 bit quantization with minimum hardware. However, it does need either a fast A/D or perferably a slow input rate from the DCT. The drawings for a possible digital implementation that has not been fully tested are given at the end of this report. Schematics for the hydrid analog/digital DPCM are available from NUC if desired.

Section VI. Frame Store Memory

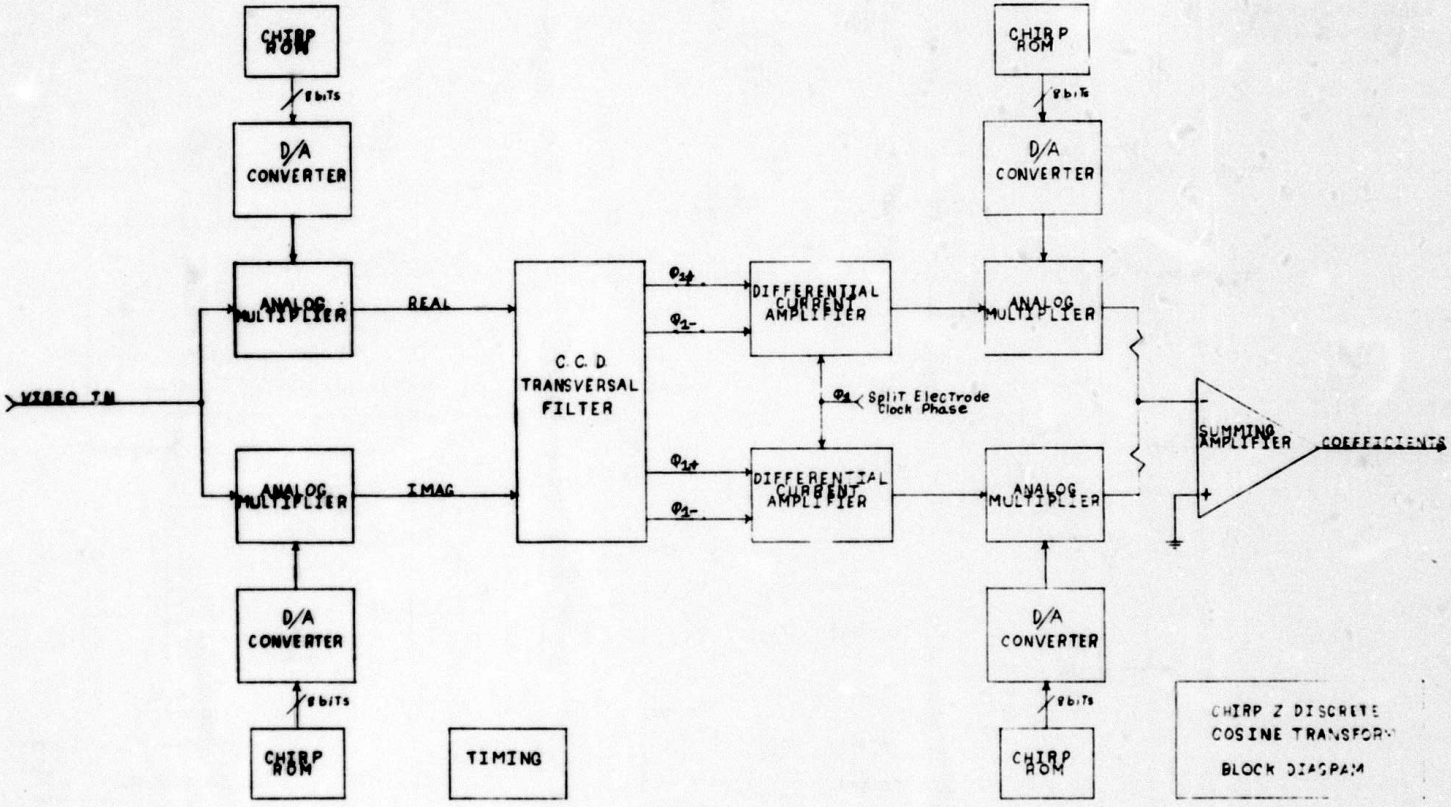
6.1 A frame store memory (FSM) is required in the ground station to reconstruct the picture from the 8 stripes that are transmitted per frame. The frame store memory as currently implemented uses a commercial 32k x 18 bit solid state memory (specs in appendix) and stores the pixels sequentally 3 pixels per 18 bit word. All reconstructed picture elements are buffered in the FSM and read out at EIA standard TV rates. Because of the 8 to 1 reduction in frame rate, each picture element is displayed 8 times before being updated.

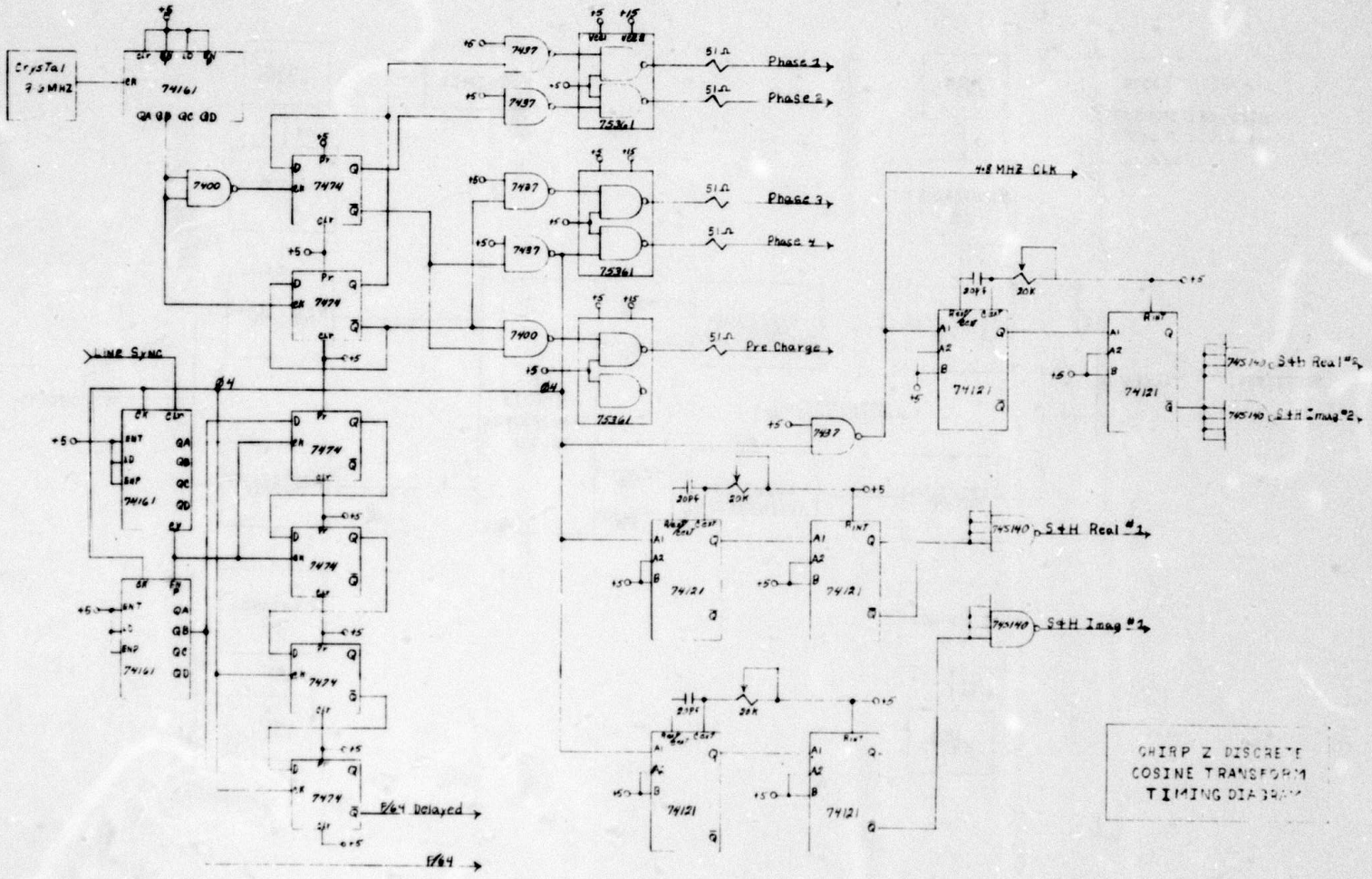
6.2 The basic operation of the frame store memory is as follows. The TV sync in the plane and on the ground are locked together as discussed in the timing and synchronization section. During the active part of each TV line, 32 picture elements (pixels) are reconstructed at the ground and buffered. Simultaneously, 256 pixels are read from the main memory and displayed. During the horizontal retrace time, the 32 new pixels are stored in the main memory. Since the camera and the display are sync-locked, the new pixels are stored in the same line as was just displayed. The stripe (1/8 line) in which the pixels are stored is determined by the stripe counter on the ground, which is also synchronous with the one in the plane.

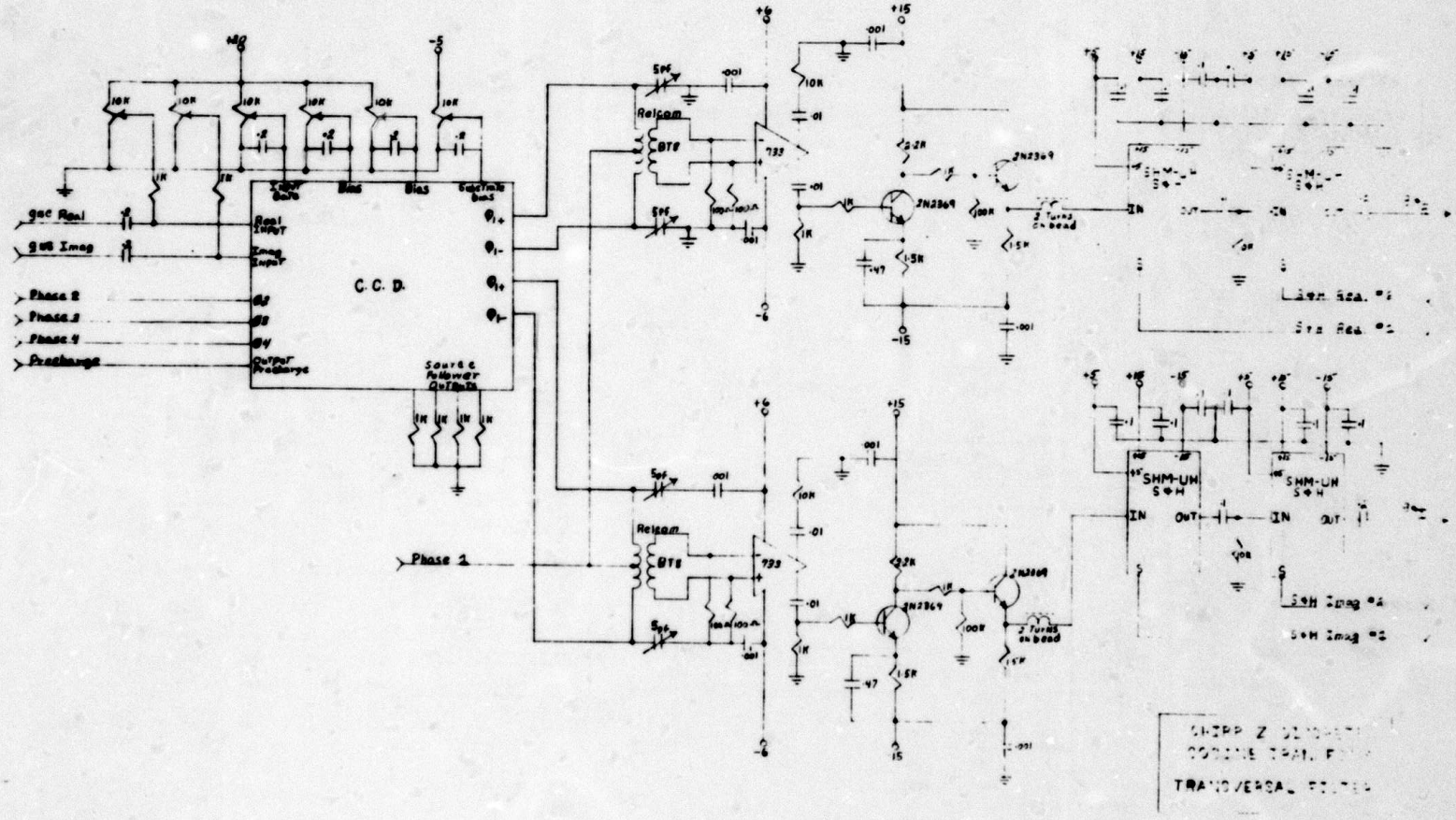
6.3 There is no distinction made anywhere in the bandwidth compression system between the even and odd fields of the TV frame. The even and odd fields are assumed to be the same, and the frame store memory stores only one field, i.e. 256 pixels x 262.5 lines. It is possible by doubling the FSM size to store both fields and thus increase the number of vertical lines to 525. There is currently, however, no desire to do this.

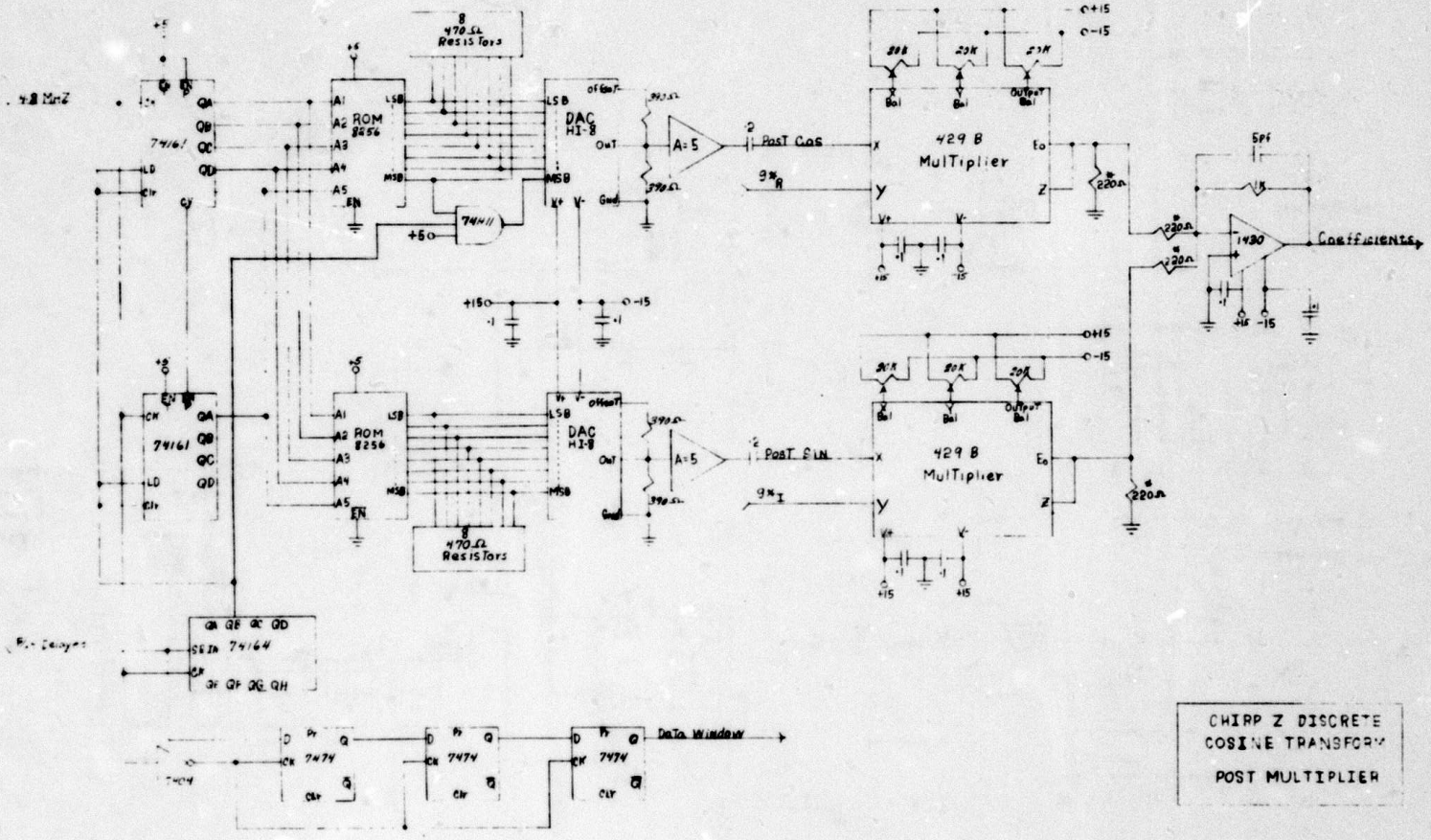
6.4 There are complications to the operation of the frame store memory due to the fact that 3 pixels are stored in each main memory word. This means that 32 pixels require 10 2/3 words. Since the line is divided into eight 32-pixel-stripes, some stripes do not start or end on even word boundaries in the main memory. A problem arises when only part of a main memory word is to be updated (i.e. 6 or 9 bits). Because that part of the word not updated is destroyed, one must compensate by reading from the main memory the pixels that could be destroyed in the update process (i.e. the first and last words of the block). The new pixels are merged with the data from the memory and rewritten, thus saving the pixels that are not updated.

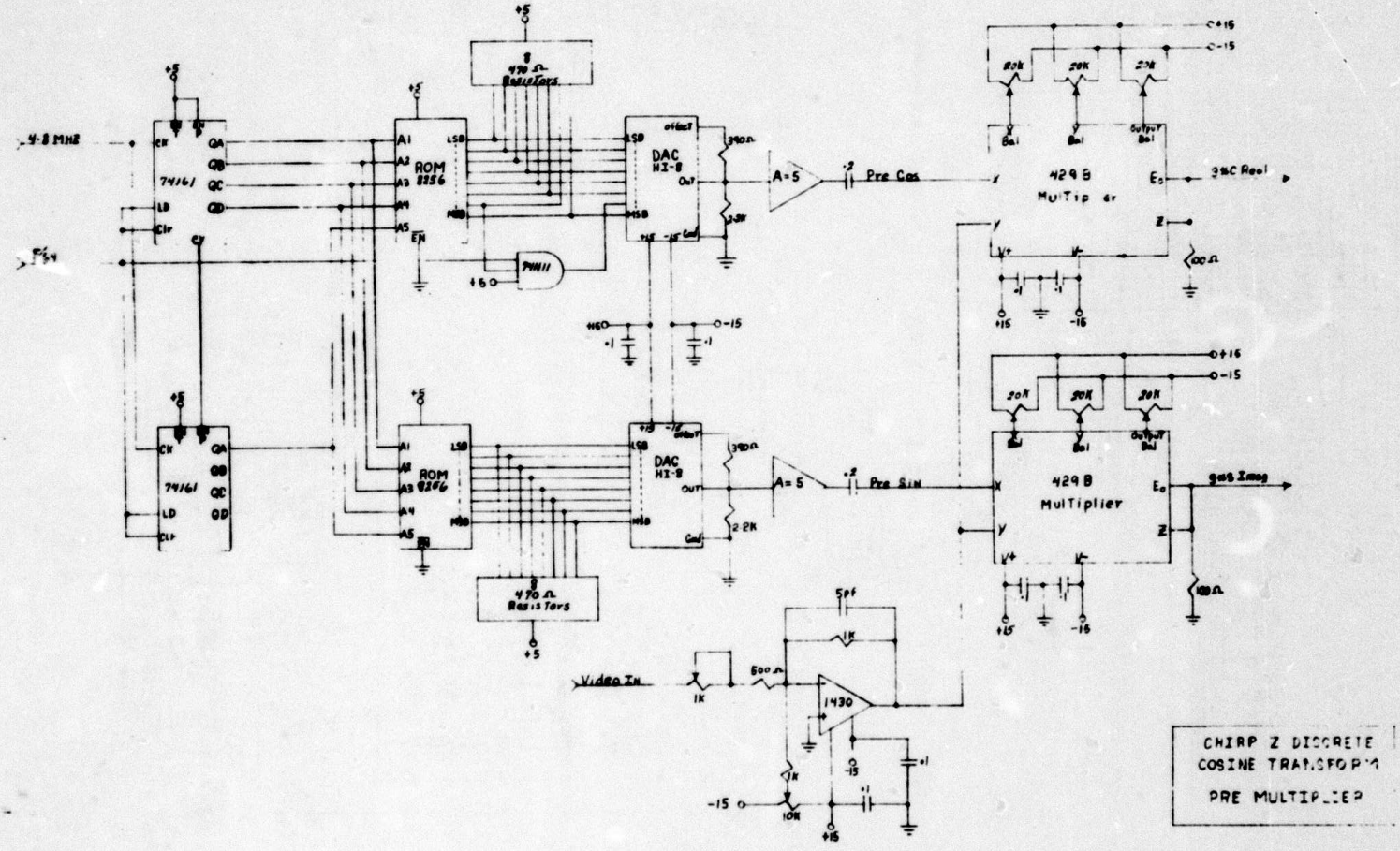
6.5 The current implementation of the FSM has only 256 lines per frame and is not set up to accept 262.5 lines per field video. The modification to the existing design to incorporate the 262.5 line video should be very minor and NUC can provide guidance in this area.

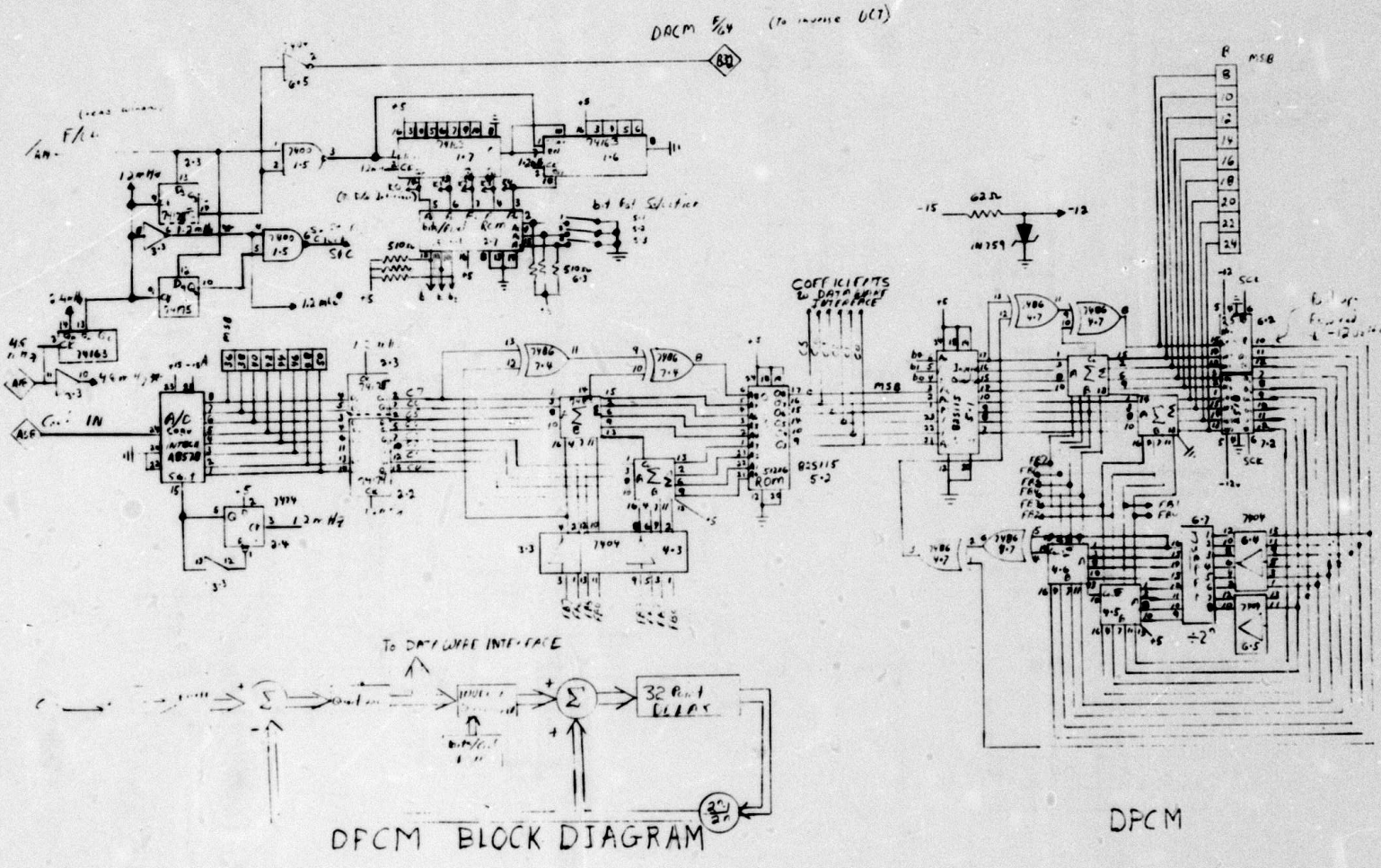

CHIRP ROM
8 bits
D/A CONVERTER
ANALOG MULTIPLIER
REAL
VIDEO IN
ANALOG MULTIPLIER
IMAG
D/A CONVERTER
8 bits
CHIRP ROM
TIMING
C.C.D. TRANSVERSAL FILTER
ϕ_{1+}
ϕ_{1-}
ϕ_{1+}
ϕ_{1-}
DIFFERENTIAL CURRENT AMPLIFIER
DIFFERENTIAL CURRENT AMPLIFIER
ϕ_1 Split Electrode Clock Phase
CHIRP ROM
8 bits
D/A CONVERTER
ANALOG MULTIPLIER
ANALOG MULTIPLIER
D/A CONVERTER
8 bits
CHIRP ROM
SUMMING AMPLIFIER
COEFFICIENTS
CHIRP Z DISCRETE COSINE TRANSFORM
BLOCK DIAGRAM

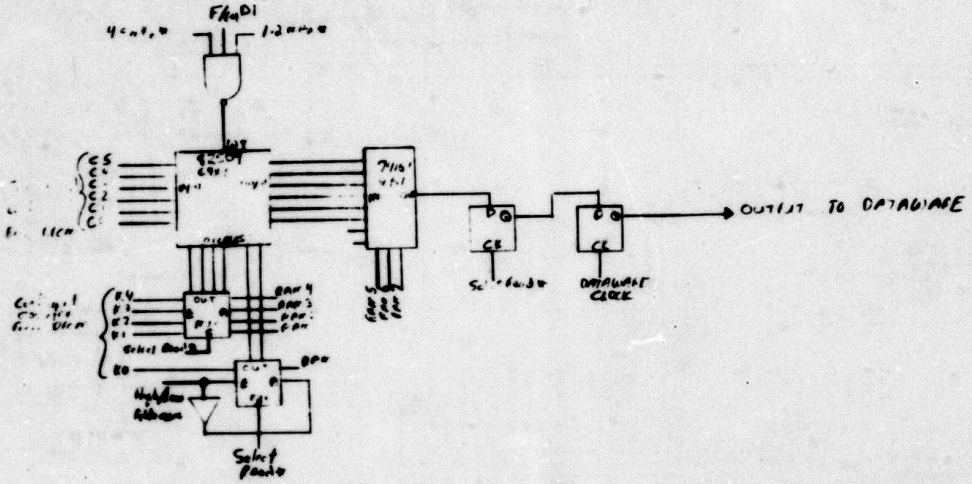

Crystal
7.2 MHz
74161
7400
7474
7437
75361
Phase 1
Phase 2
Phase 3
Phase 4
Pre Charge
51Ω
LINE SYNC
Ø4
F/64 Delayed
F/64
4.8 MHZ CLK
74121
745140
S+H Real #2
S+H Imag #2
S+H Real #1
S+H Imag #1
CHIRP Z DISCRETE
COSINE TRANSFORM
TIMING DIAGRAM

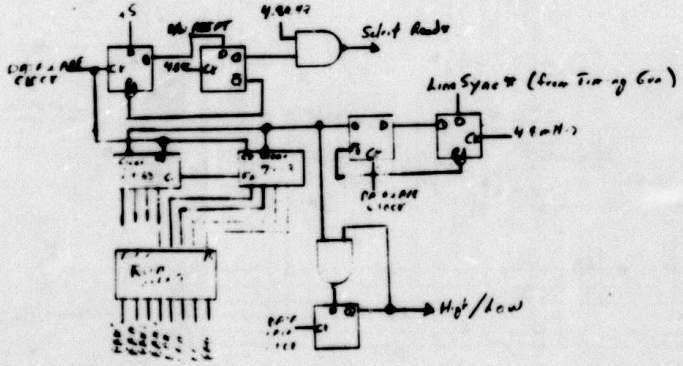

C. C. D.
Phase 1
Phase 2
Phase 3
Phase 4
Precharge
Relcom
2N2369
SHM-UH
S+H
IN
OUT
TRANSVERSAL FILTER


8 470Ω Resistors
ROM 8256
DAC HI-8
74161
74164
74H11
7404
7474
A=5
429 B MulTiplier
1430
Post Cos
Post Sin
Coefficients
Data Window
CHIRP Z DISCRETE COSINE TRANSFORM
POST MULTIPLIER

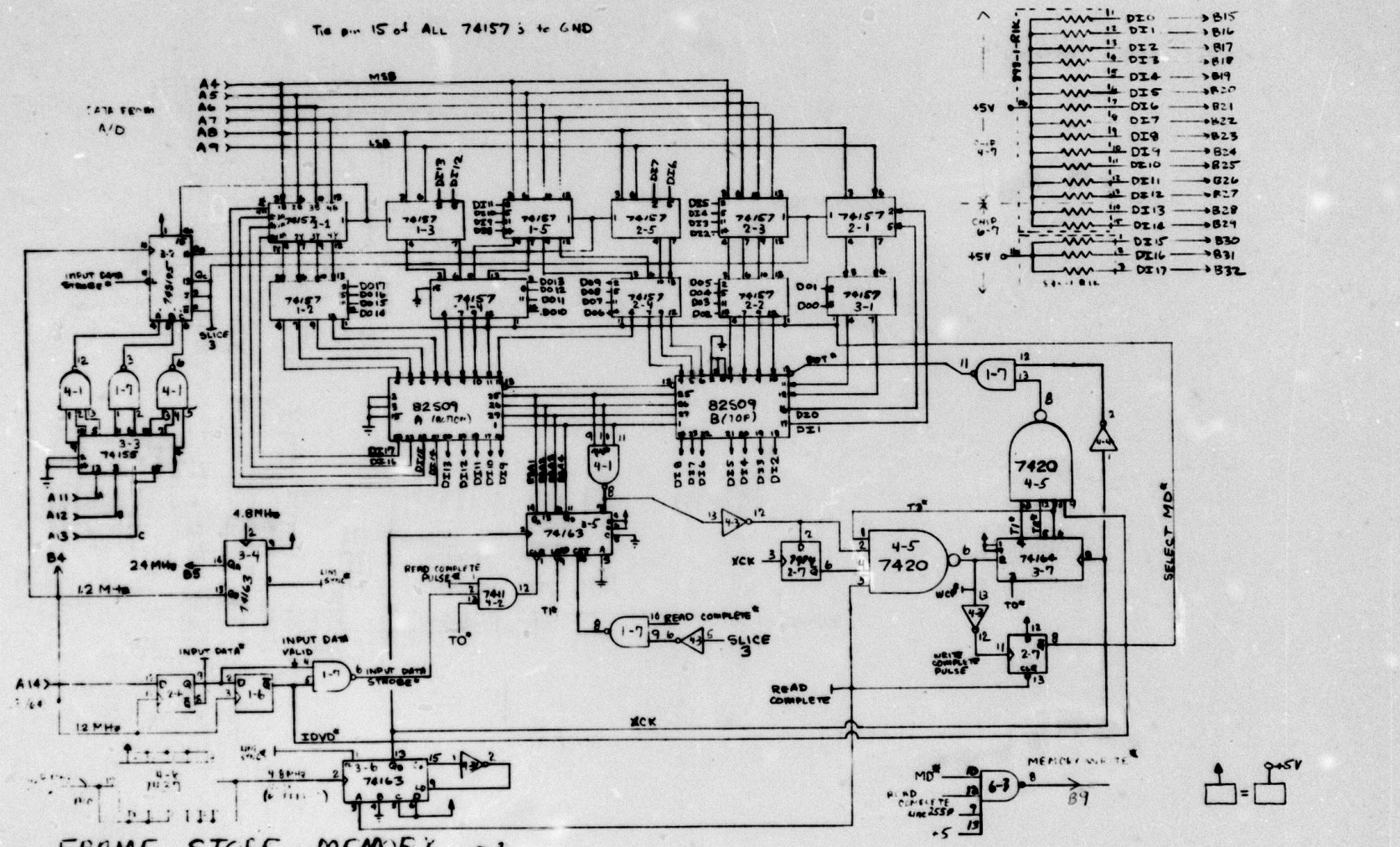

4.8 MHz
74161
ROM 8256
DAC HI-8
PWM
470 Ω Resistors
A=5
Pre Cos
Pre Sin
429B Multiplier
Video In
1430
CHIRP Z DISCRETE COSINE TRANSFORM
PRE MULTIPLIER


DACM F/64 (To inverse DCT)
COEFFICIENTS TO DATA ADAPT INTERFACE
MSB
To DATA WAVE INTERFACE
32 Point FIFO
DPCM BLOCK DIAGRAM
DPCM

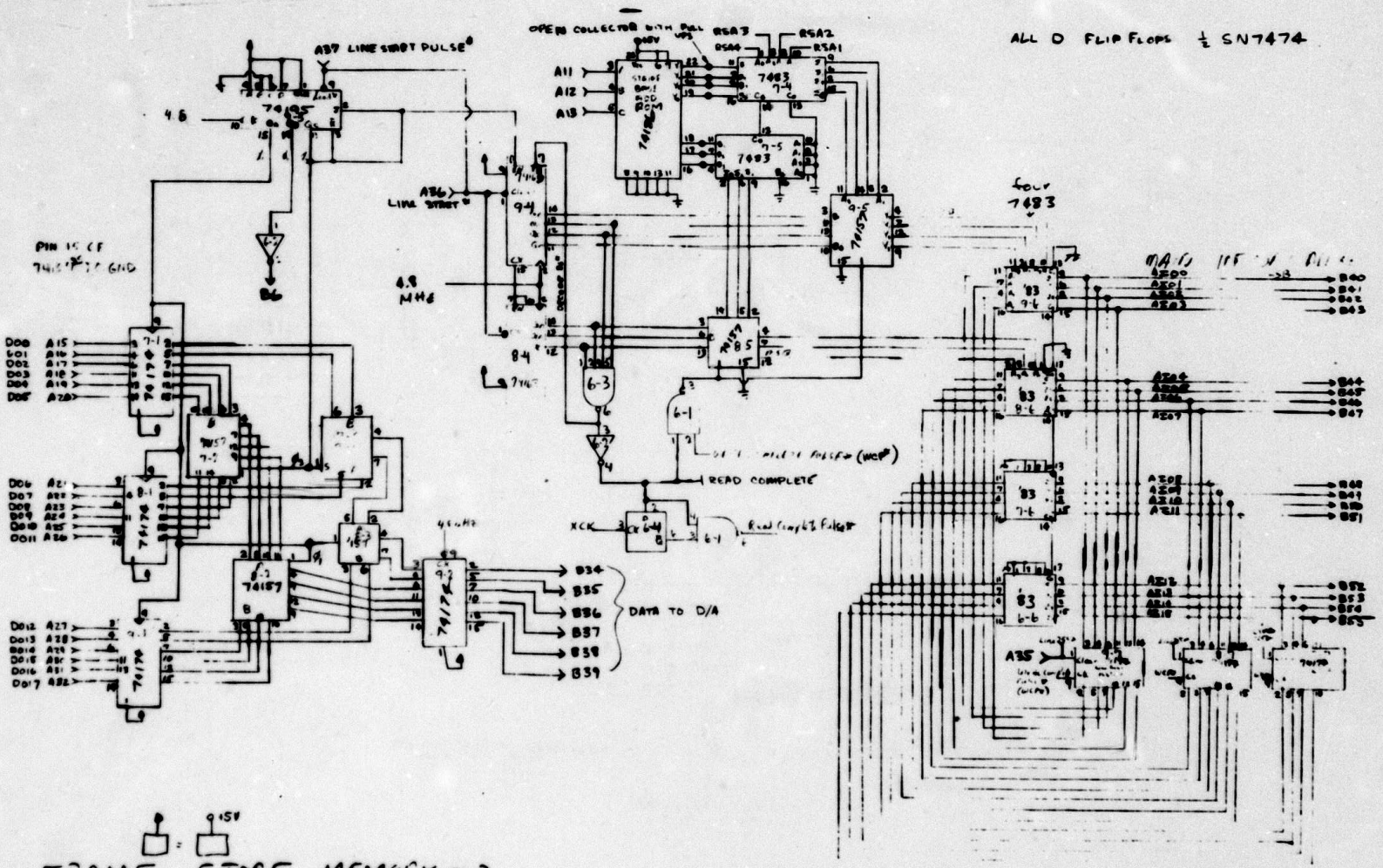




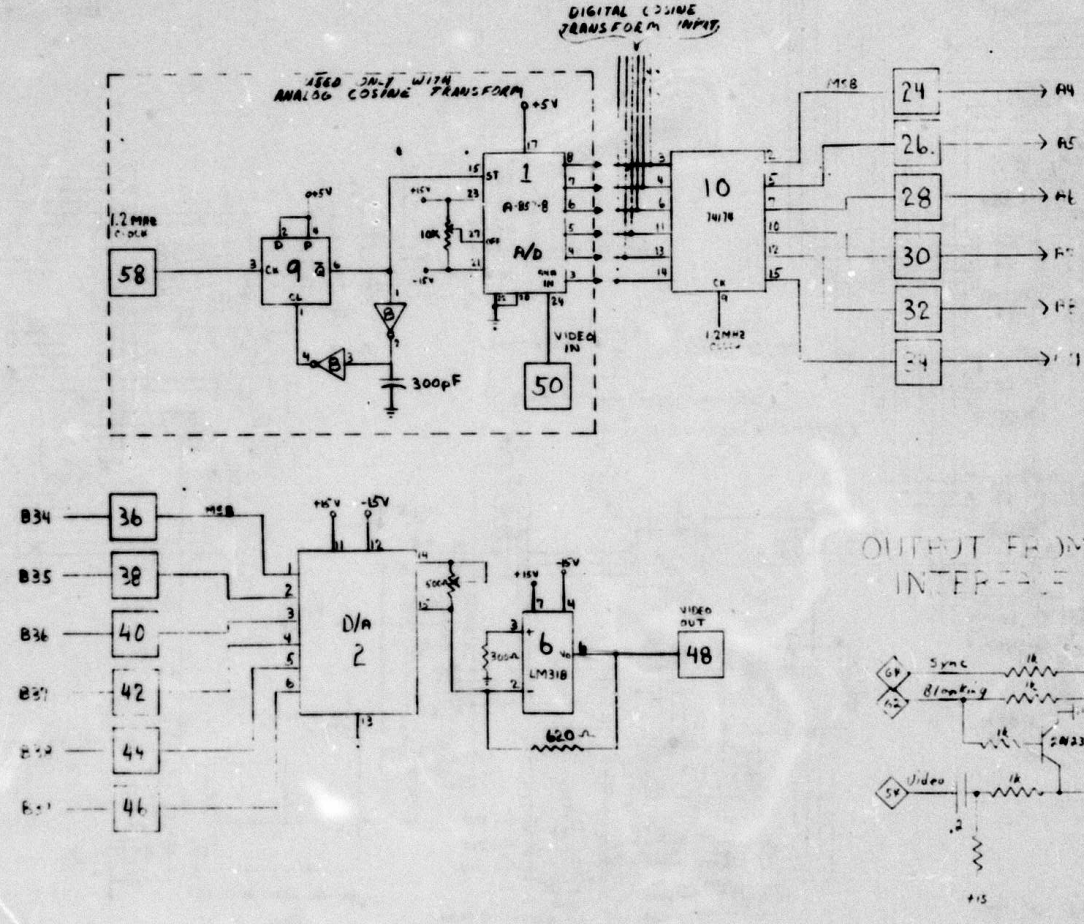
DATABUFFER CODED ZOOMFACT

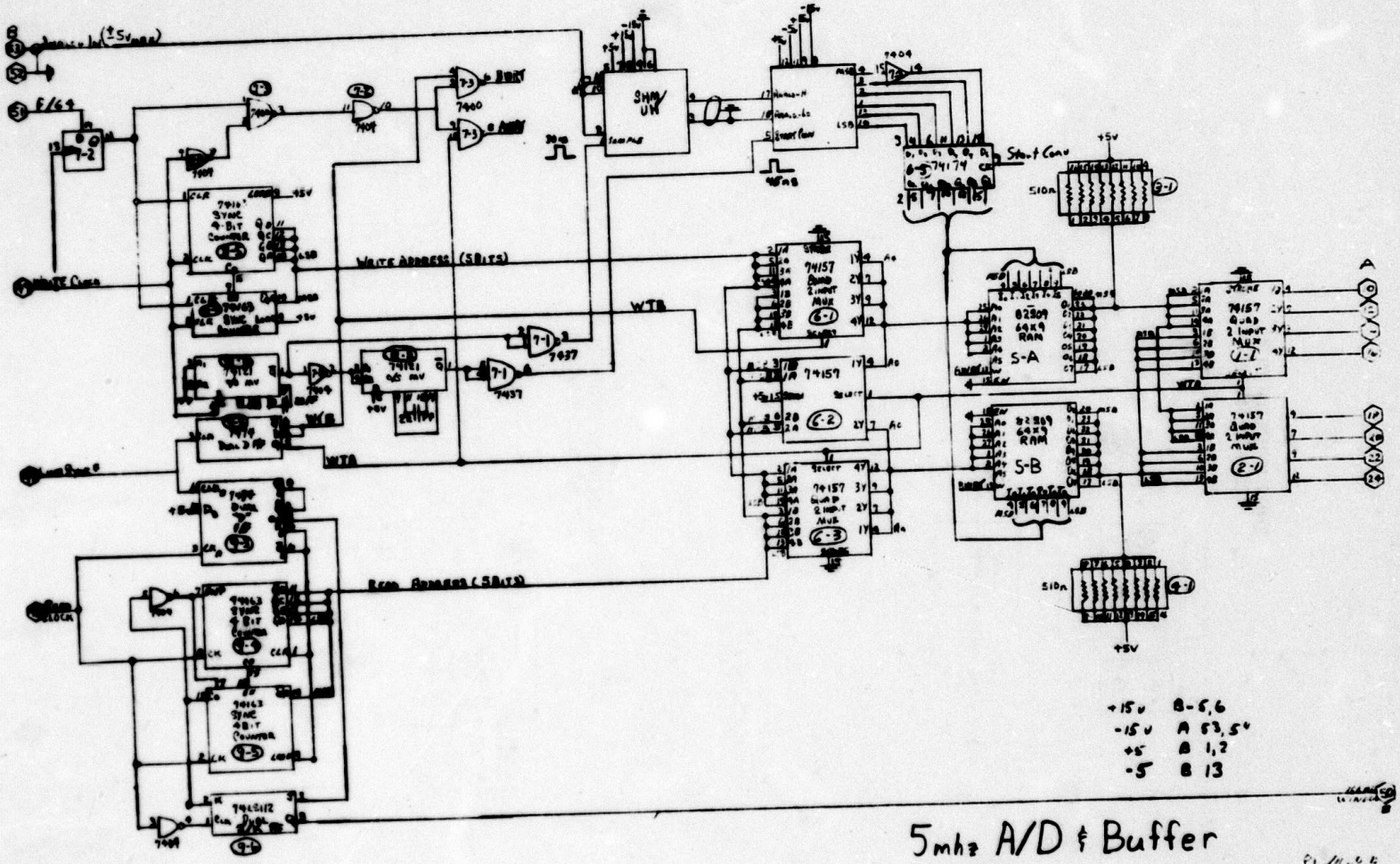


FRAME STORE MEMORY #1



FRAME STORE MEMORY pg 2.


DIGITAL COSINE TRANSFORM INPUT
VIDEO IN-T WITH ANALOG COSINE TRANSFORM
1.2 MHz CLOCK
58
+5V
+15V
-15V
10K
A/D
VIDEO IN
50
300pF
1.2MHz
10
74174
MSB
24
26
28
30
32
34
A4
A5
A6
A7
A8
B34
B35
B36
B37
B38
36
38
40
42
44
46
MSB
D/A 2
+15V
-15V
5KΩ
300Ω
LM318
620Ω
VIDEO OUT
48
OUTPUT FROM
INTERFACE
Sync
Blanking
Video
100Ω
1K
SN2369
3/8
+5
-15
.2
100Ω



5mhz A/D & Buffer

(12-75)

DIRECT IMPLEMENTATION OF
DCT BLOCK DIAGRAM

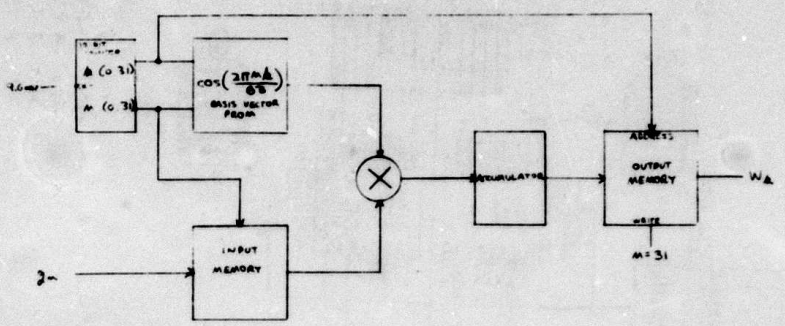


FIGURE 3.4

ARC-COSINE IMPLEMENTATION OF
DCT BLOCK DIAGRAM

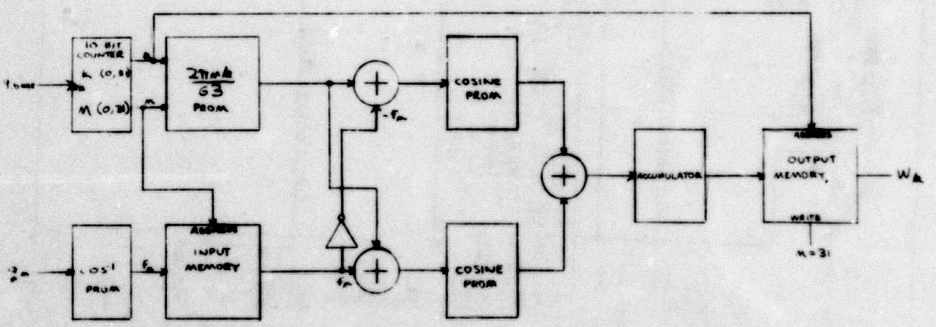
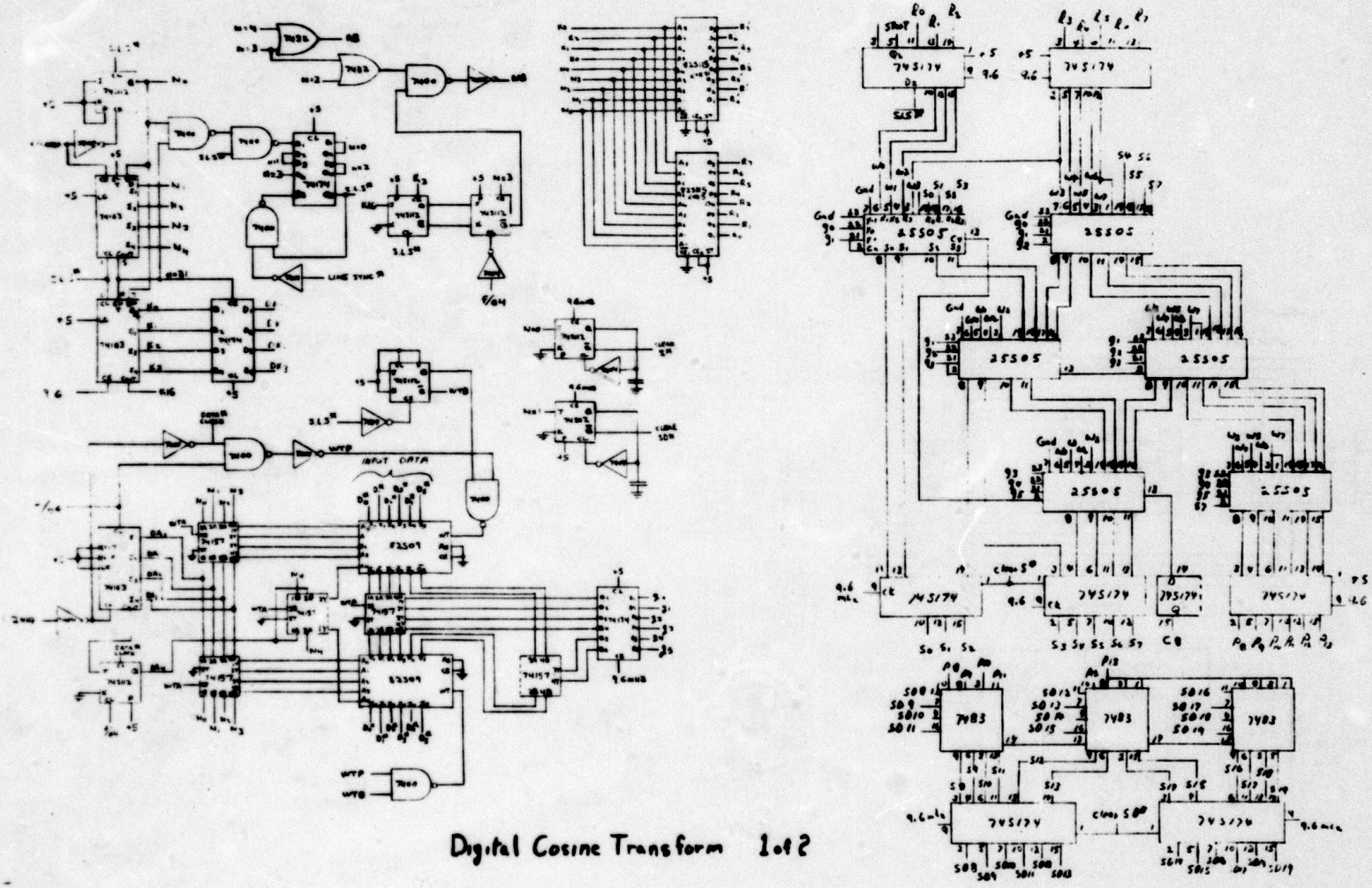
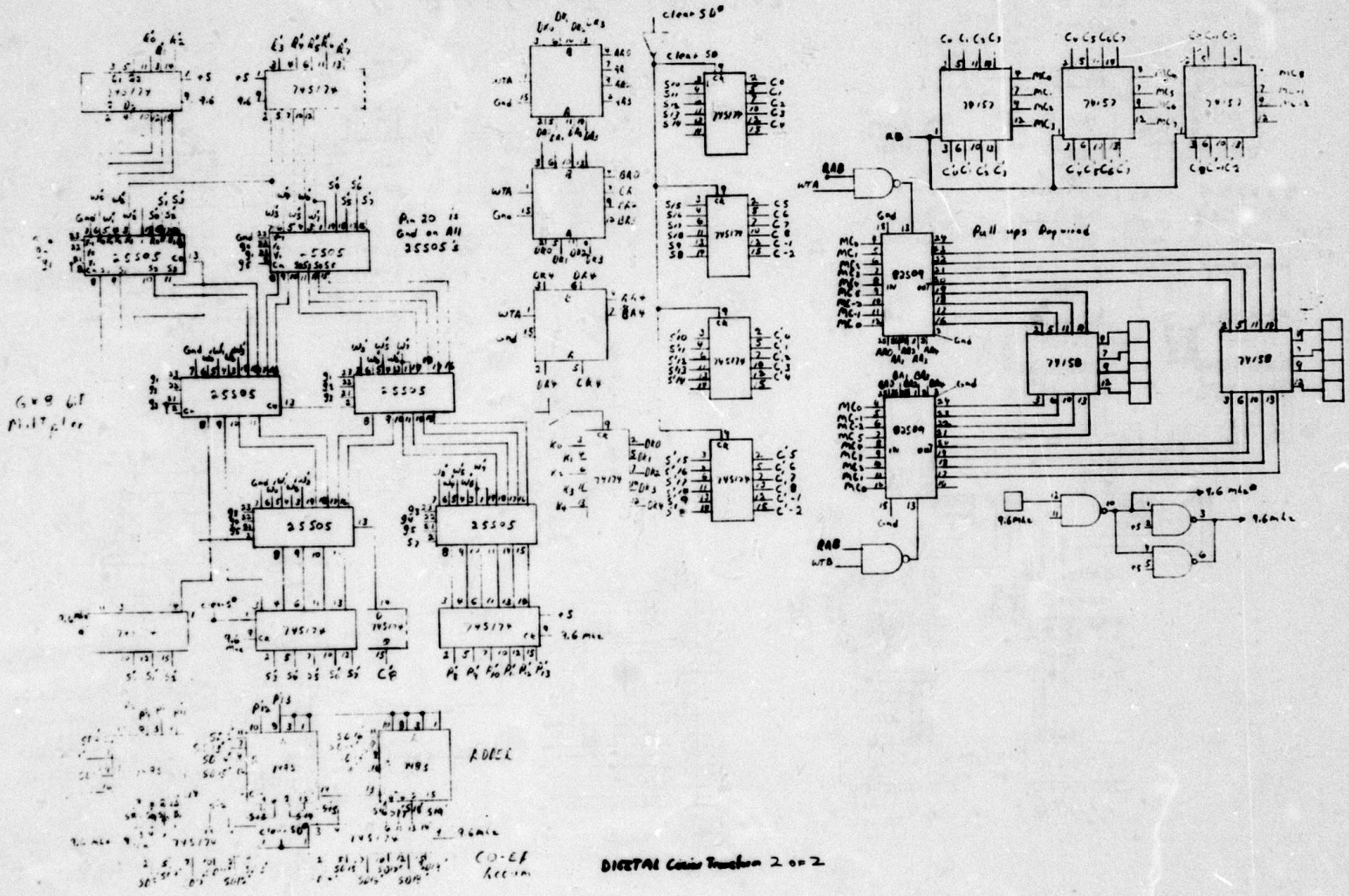


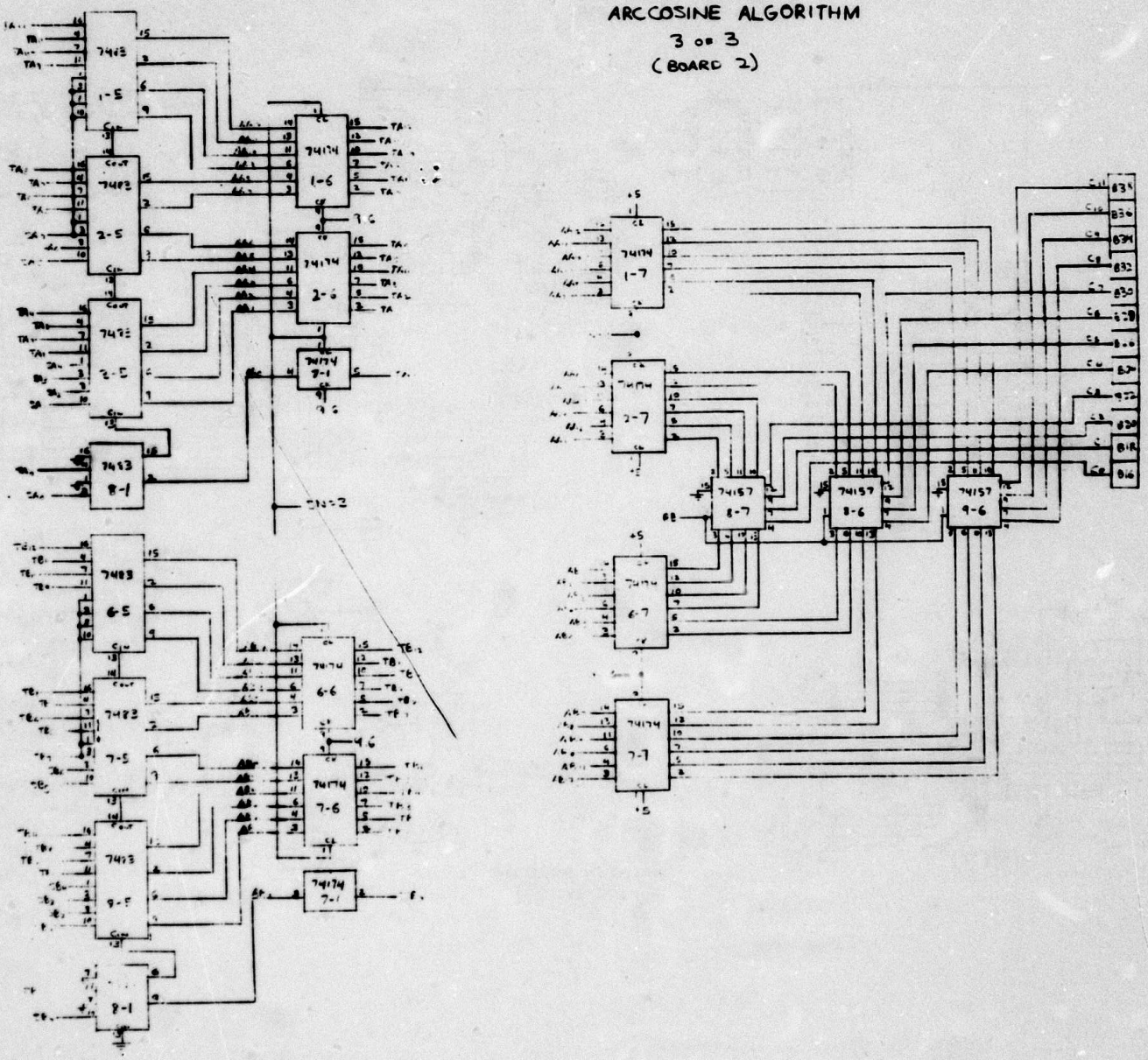
FIGURE 3.4



Digital Cosine Transform 1 of 2

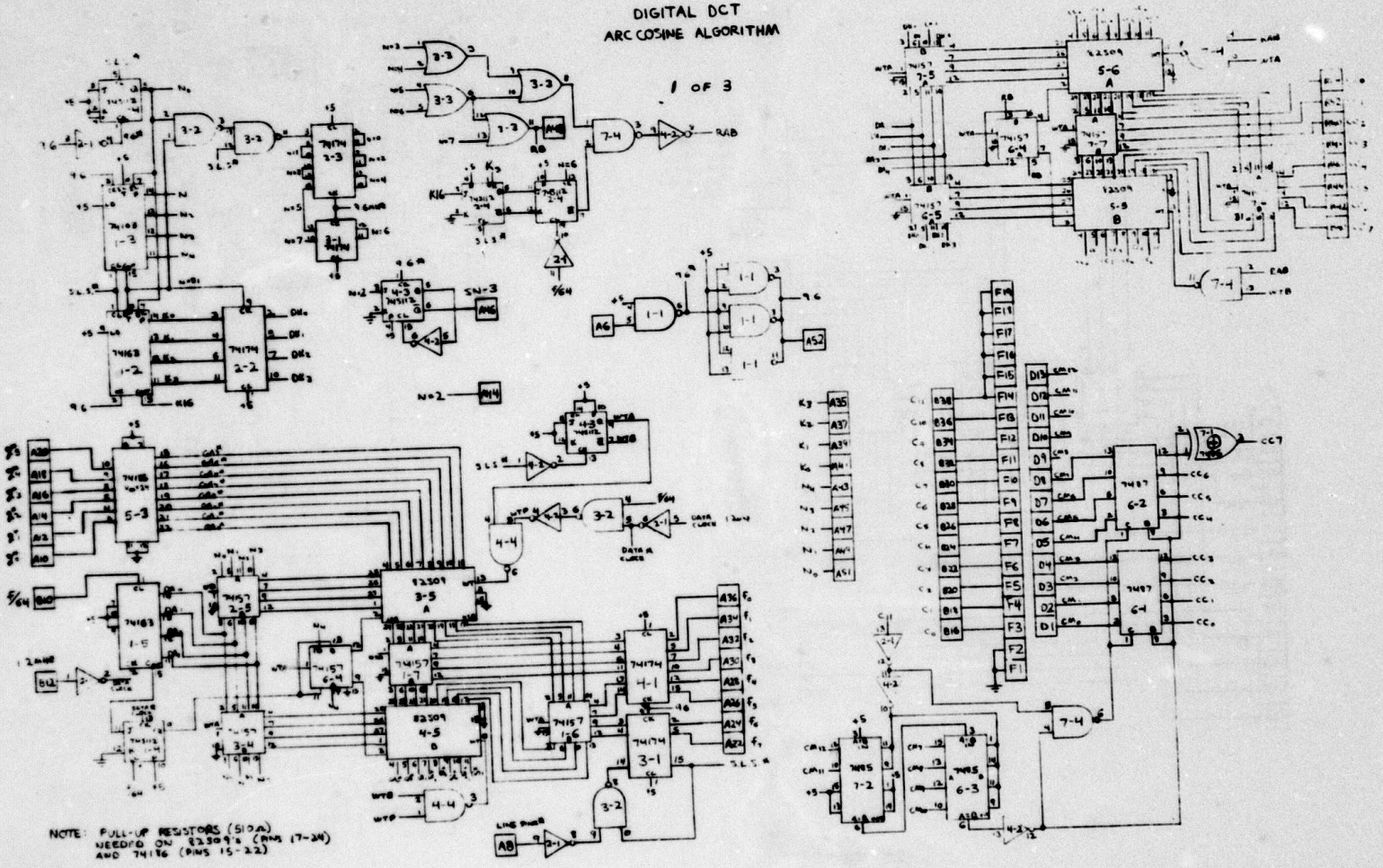

Pin 20 is Gnd on All 25505's
Pull ups Required
DIGITAL Comic Translator 2 of 2

DIGITAL DCT
ARCCOSINE ALGORITHM
3 of 3
(BOARD 2)

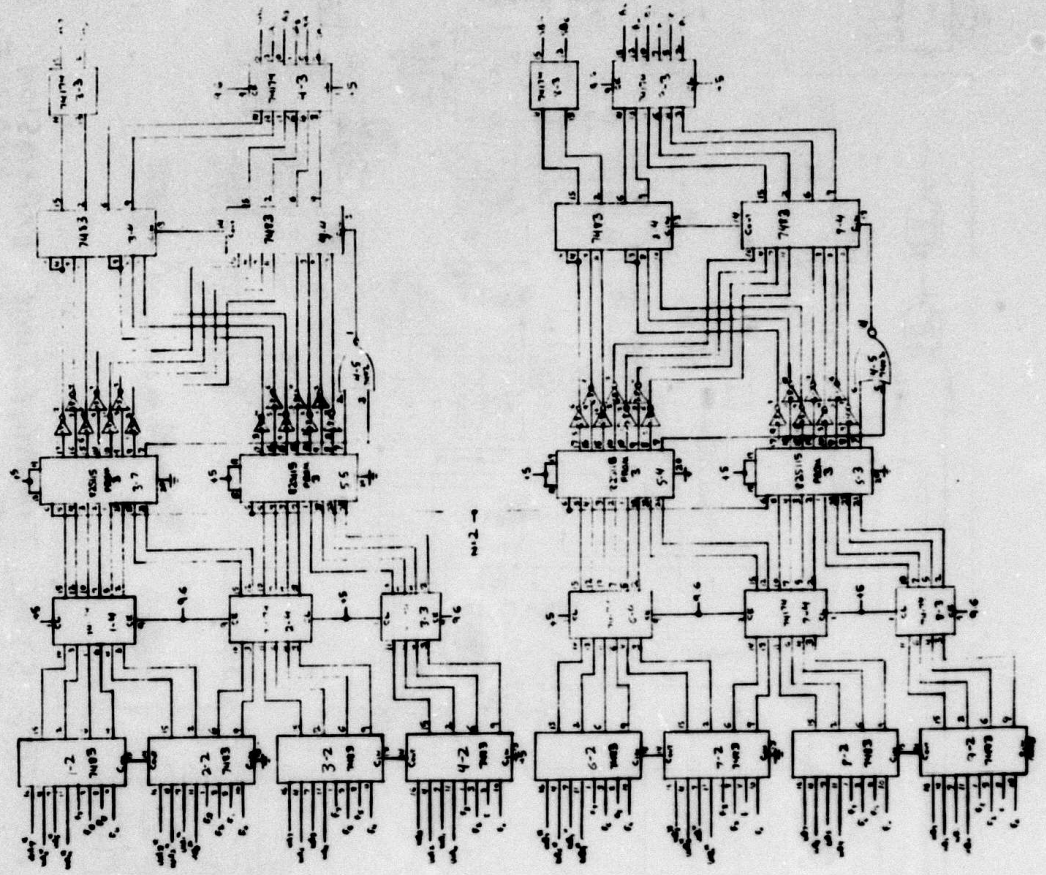


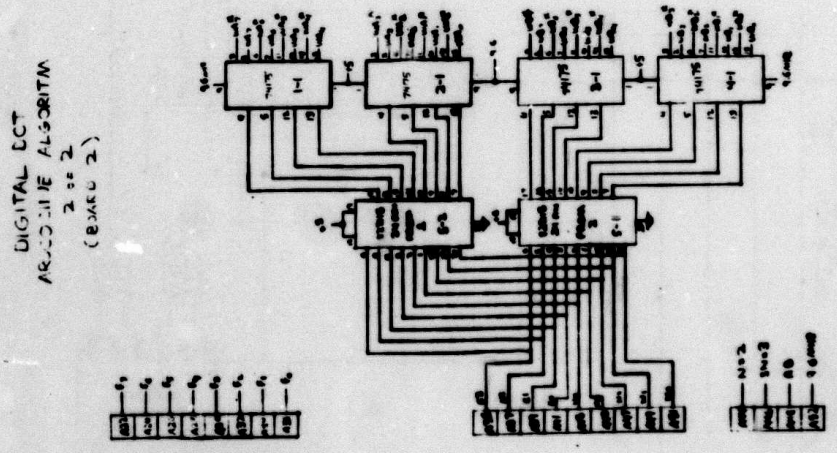
DIGITAL DCT
ARC COSINE ALGORITHM

1 OF 3

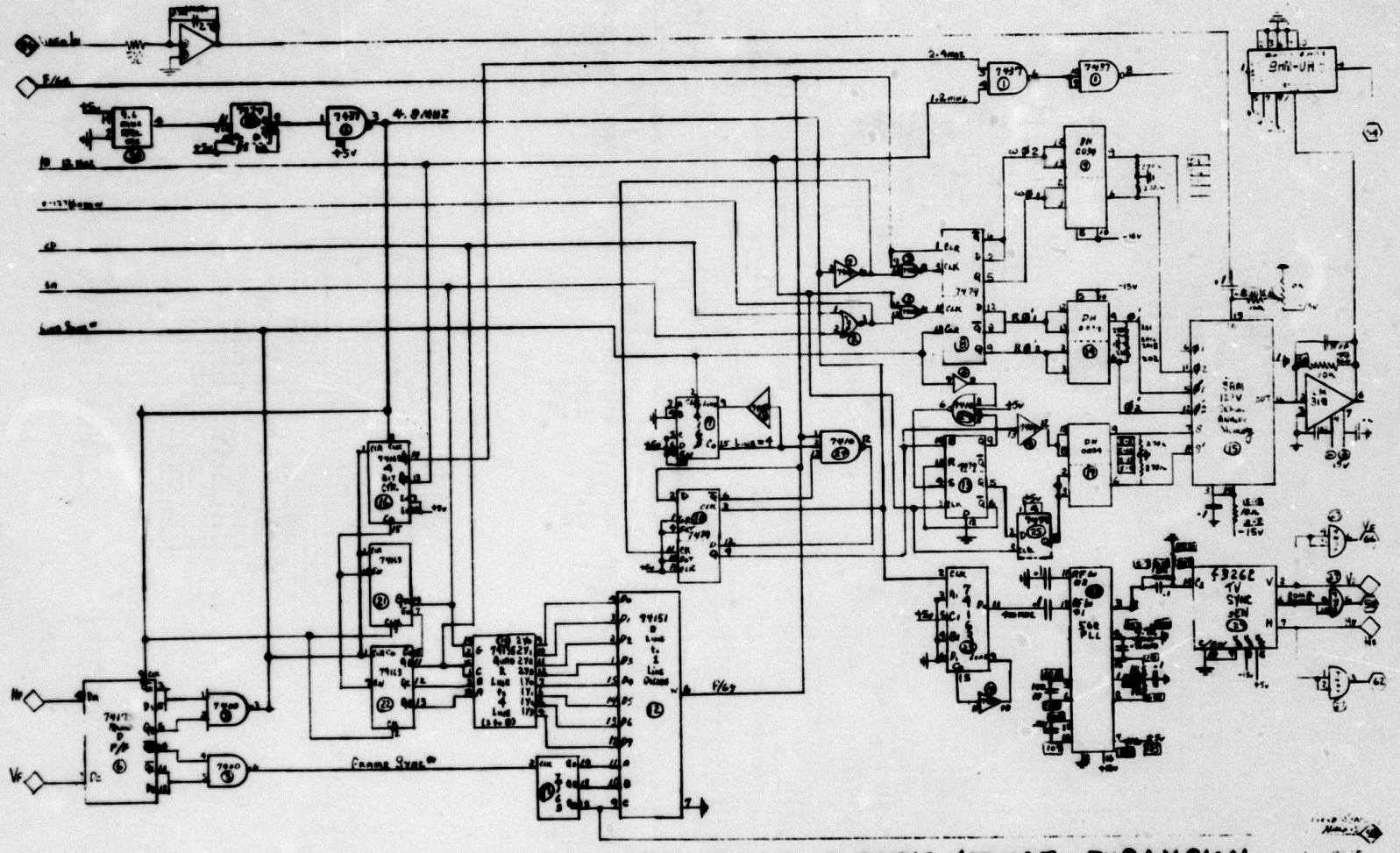


NOTE: PULL-UP RESISTORS (510Ω) NEEDED ON 82309's (PINS 17-20) AND 74196 (PINS 15-22)





DIGITAL DCT
ARCHITECTURE ALGORITHM
2 OF 2
(BASED 2)



TV SYNC, TIMING, & TIME EXPANSION MODULE

APPENDIX B

COMBINED SPATIAL AND TEMPORAL CODING OF DIGITAL IMAGE SEQUENCES

COMBINED SPATIAL AND TEMPORAL CODING OF DIGITAL IMAGE SEQUENCES*

John A. Roese
Naval Undersea Center
San Diego, California 92132

Guner S. Robinson
Image Processing Institute, University of Southern California
Los Angeles, California 90007

Abstract

Interframe coding of television images encompasses techniques which make use of correlations between pixel amplitudes in successive frames. Intraframe coding techniques that exploit spatial correlations can, in principle, be extended to include correlations in the temporal domain.

In this paper, successive frames of digital images are coded using two-dimensional spatial transforms combined with DPCM in the temporal domain. Specific transform techniques investigated are the two-dimensional cosine and Fourier transforms. Due to DPCM encoding in the temporal domain, the hybrid transform/DPCM encoders require storage of only the single previous frame of data.

Hardware implementation of the Fourier transform involves manipulation of complex numbers where the cosine transform does not. However, the Fourier transform is attractive because frame-to-frame motion compensation can be introduced directly in the phase plane by application of appropriate phase correction factors.

Results are presented in terms of coding efficiency, storage requirements, computational complexity, and sensitivity to channel noise.

Introduction

In the design of image coding systems for digital communications channels, the primary objective is to minimize the number of code bits required to reconstruct the image at the receiver. Reduction in the number of code bits transmitted results in reduced channel bandwidth, more rapid transmission of digital images, and lower transmitter power requirements.

Efficient coding of the digital images is accomplished by removal of statistical redundancies that exist within the image. Transform, predictive, and hybrid transform/predictive image coding techniques have been developed to exploit intraframe spatial image redundancies. This paper describes efforts to extend these image coding techniques to coding of time-sequences of digital images transmitted over a digital communications channel. The emphasis has been directed towards definition of an image coding system that exploits temporal as well as spatial image redundancies.

Intraframe Image Coding

The primary techniques that have been developed for intraframe image coding in the spatial domain are transform, linear predictive, and hybrid transform/linear predictive techniques. Operational descriptions of these coding methods are given below.

Transform Image Coding

The basic premise of transform image coding is that the transform domain representation of an image has an energy distribution that is more compact and therefore easier to efficiently code than the spatial domain version. In transform coding systems, a one- or two-dimensional linear transform of an image line segment or block is performed at the coder. The transform coefficient statistics are computed prior to quantization and coding for transmission. After decoding at the receiver, an inverse transform is taken to obtain a reconstructed image. Transforms that have proven useful for this application include Fourier, cosine, Hadamard, Slant, and Karhunen-Loeve.[1,2,3] It should be noted that the two transforms can be different for the two spatial directions.

Two-dimensional transforms have the inherent disadvantage of requiring an intermediate memory to store the transform coefficients computed in one direction while the transform is being computed in the other direction. Transform coding techniques have been explored extensively both theoretically and by simulation. It has been shown that significant bit rate reductions can be achieved in many applications with minimal image degradation.

*To be published in the SPIE Conference Proceedings (Aug. 1975).

Simulation results indicate that a bit rate reduction to 1.5 bits/pixel can be obtained for monochrome image transform coding in 16 x 16 picture element (pixel) blocks, while color images require about 2.0 bits/pixel.[4] The bit rate can be reduced further by making the transform coding system adaptive.

Predictive Image Coding

The high degree of correlation between a given pixel value and its nearest neighbors allows an image to be efficiently represented by coding only the difference between each pixel value and its predicted value. The predicted pixel value is based on previously scanned pixel values. In a differential pulse code modulation (DPCM) system, the predicted value of each pixel is subtracted from its actual value and this difference is quantized and transmitted. Quantization of the data is done with a quantizer designed for the probability density of the difference signal. Thus, coding by use of a DPCM system requires a knowledge of the data statistics. The basic operation of the DPCM coder is to generate an uncorrelated signal which is then encoded by a memoryless quantizer for transmission. At the receiver, the quantized difference signal is combined with its predicted value to form the reconstructed pixel value. Basic DPCM image coding systems provide good quality at about 3 bits/pixel. Adaptive systems, in which parameters of the quantizer and predictor adapt to the image content, require about 2 bits/pixel.[5]

Hybrid Image Coding

Analysis of transform and predictive coding techniques has shown that both techniques possess attractive characteristics and certain limitations. Transform coding techniques achieve good performance at low bit rates, show less sensitivity to data statistics (picture-to-picture variations) and are less vulnerable to channel noise. Predictive coding systems are superior to transform techniques in terms of equipment complexity, memory requirements, and performance at high bit rates. Some limitations of predictive systems are their sensitivity to data statistics and to channel error.

An intraframe hybrid coding system that combines the attractive features of both transform and predictive coding systems has been devised.[6] In this system, a one-dimensional transform is followed by DPCM linear predictive coding of the transform domain coefficients. At the receiver, the transform coefficients are decoded and a replica of the original image is reconstructed by an inverse transform.

Hybrid Interframe Image Coding

Interframe coding of digital image sequences encompasses those techniques which make use of the high correlation that exists between pixel amplitudes in successive frames. Intraframe coding techniques that exploit spatial correlations can, in principle, be extended to include correlations in the temporal domain. Previous research in the area of three-dimensional Fourier and Hadamard transformations has indicated that bit rates can be reduced by a factor of five by incorporating correlations in the temporal direction.[7] However, three-dimensional transform systems are unattractive as they use large amounts of data storage and require excessive computations.

To alleviate the problems associated with three-dimensional transform systems, new hybrid (two-dimensional transform)/DPCM image coding systems have been developed.[8] These systems utilize both spatial and temporal correlations while greatly reducing memory storage and computational requirements. The block diagram for a hybrid (two-dimensional transform)/DPCM system is shown as figure 1. In present implementations of this system, either a two-dimensional cosine or Fourier transformation is performed on 16 x 16

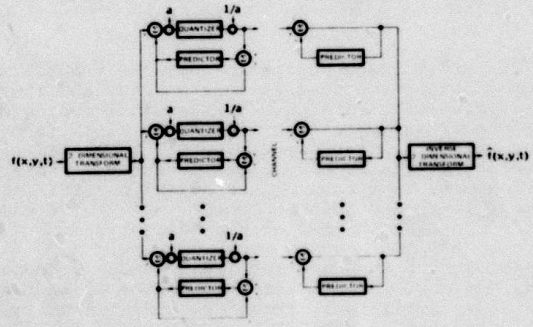


Fig. 1. Hybrid (two-dimensional transform)/DPCM coder.

subblocks. DPCM linear predictive coding in the temporal domain is then applied to the transform coefficients of each subblock. For notational convenience, the hybrid interframe coders employing two-dimensional Fourier transforms will be denoted as FFD and those using two-dimensional cosine transforms as CCD. The FFD and CCD coders are adaptive in the sense that statistics of the transform coefficient differences of each subblock are computed prior to encoding the transform coefficients in the temporal direction by parallel DPCM coders. At the receiver, the transmitted transform coefficients are decoded and a replica of each frame is reconstructed by the appropriate inverse two-dimensional transformation. These systems require only a single frame of storage and involve significantly less memory and fewer computations than three-dimensional transform coding techniques.

Operational Modes

At least three operational modes have been identified for the hybrid interframe coding systems. These operational modes depend on the initial conditions assumed for the predictive coder. The initial conditions are:

a. No apriori information available at the receiver,

b. Limited information (such as mean, variance and temporal correlations based on a statistical model) available at the receiver, and

c. First frame available at the receiver.

In the no apriori information available case, several frames are required for the hybrid coder to settle. However, it has been experimentally verified that in the remaining two cases, nearly stable coder performance is achieved within the first 4 to 6 frames. From operational considerations, the third set of initial conditions is the most realistic as periodic full frame updating will be required to eliminate the cumulative effects due to channel noise.

Mathematical Formulation

Let $f(x,y)$ denote a two-dimensional array of intensity values on an NxN subblock of a digital television image of size MxM. Typical values for M and N are 256 and 16, respectively. Let $F(u,v)$ be the two-dimensional array obtained by taking the two-dimensional transform of $f(x,y)$. In the case of the two-dimensional discrete Fourier transform, the expressions relating $f(x,y)$ and $F(u,v)$ are

$$F(u,v) = \frac{1}{N^2} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \exp\left[-\frac{2\pi i}{N}(ux + vy)\right] \qquad (1)$$

$$u,v = 0, 1, \ldots, N-1$$

and

$$f(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u,v) \exp\left[+\frac{2\pi i}{N}(ux + vy)\right] \qquad (2)$$

$$x,y = 0, 1, \ldots, N-1$$

where N is the size of the square subblock, $f(x,y)$ is the image intensity function in the spatial domain, x and y are spatial coordinates, $F(u,v)$ is the Fourier transform, and u and v are spatial frequencies.

For image processing applications, $f(x,y)$ is a positive real function representing brightness of the spatial sample. The two-dimensional Fourier transform of a real-valued function has the following conjugate symmetry property:

$$F^*(u,v) = F(N-u,N-v) \quad , \quad u,v = 1, 2, \ldots, \frac{N}{2} - 1$$

The Fourier transform consists of $2N^2$ components, i.e., the real and imaginary or magnitude and phase components of each spatial frequency. However, as a result of the conjugate symmetry properties mentioned above, only N^2 components are required to completely define the Fourier transform.[9]

In the case of the Fourier transform, a shift in the spatial-domain variables results in a multiplication of the Fourier transform of the un-shifted image by a phase factor. If the input image $f(x,y,t_1)$ is shifted by the amount x_0 in the x-direction and y_0 in the

y-direction between times t_1 and t_2, then the Fourier transform of the shifted image is given by

$$F(u,v,t_2) = F(u,v,t_1) \exp\left|\frac{2\pi i}{N}(ux_0 + vy_0)\right| \tag{3}$$

This shifting property is expected to be useful in detecting and compensating for effects of motion between frames since many types of motions such as panned motion produce significant changes in phase components and small changes in amplitude components. Thus, compensation for platform motion may be implemented directly in the array of phase components by application of appropriate phase correction factors.

The two-dimensional Fourier transform F(u,v) of a spatial signal function f(x,y) is separable, i.e., it can be computed as two sequential one-dimensional transforms since the Fourier kernel,

$$\exp\left|\pm\frac{2\pi i}{N}(ux + vy)\right| ,$$

is separable and symmetric. Thus, the basic one-dimensional discrete Fourier transform that must be performed is

$$F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x) \exp\left(-\frac{2\pi i}{N} ux\right) \tag{4}$$

for u=0, 1, . . ., N-1.

In the case of the discrete Cosine transform[3], the one-dimensional transform is

$$F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x) \cos\frac{(2x+1)u\pi}{2N} \tag{5}$$

for u=0, 1, . . ., N-1. The form of eq. (5) differs from that of reference [3] only by a normalization constant. The cosine transform is also separable and a two-dimensional discrete cosine transform of an NxN subblock results in N^2 real coefficients.

Quantization

Experimental evidence derived from transmission of a typical "head and shoulders" picture telephone scene has shown that the frame difference signal has a probability density closely approximated by a double sided exponential function.[10] The optimum minimum mean square error quantizer for this distribution has been found to be a uniform quantizer combined with a companding of the frame difference signal.[11]

Since the variances of the transform domain coefficient differences are different, it is necessary to use different quantizer parameters for each one. Each coefficient difference signal is allocated a number of bits proportional to its estimated variance in accordance with an optimum bit assignment algorithm.

Fidelity Criteria

In figure 1, differences between input signal f(x,y,t) and output signal $\hat{f}(x,y,t)$ are due to two sources: quantization errors and errors due to channel noise. To evaluate coding efficiency of the hybrid encoders, two objective criteria were used. The first criterion, NMSE, is a measure of the mean square error between f(x,y,t) and $\hat{f}(x,y,t)$ averaged over an entire frame of size MxM. Normalization is achieved by dividing the mean square error by the mean signal energy within the frame.

$$\text{NMSE} = \frac{\frac{1}{M^2}\sum_{x=0}^{M-1}\sum_{y=0}^{M-1}\left|f(x,y,t) - \hat{f}(x,y,t)\right|^2}{\frac{1}{M^2}\sum_{x=0}^{M-1}\sum_{y=0}^{M-1}\left|f(x,y,t)\right|^2} \tag{6}$$

The second criterion, SNR, measures the ratio of peak-to-peak signal to RMS noise:

$$SNR = -10 \log_{10} \frac{\frac{1}{M^2} \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} \left| f(x,y,t) - \hat{f}(x,y,t) \right|^2}{255^2} \qquad (7)$$

Figures 2 and 3 are graphs illustrating the coding efficiency of the hybrid FFD and CCD coders at various bit rates in the interval 0.1 to 1.0 bits/pixel/frame. To perform this series of experiments, a 256 x 256 resolution data base consisting of 16 consecutive frames of a 24 frames per second (fps) motion picture was digitized. Initial conditions assumed were that the first frame was available at the receiver.

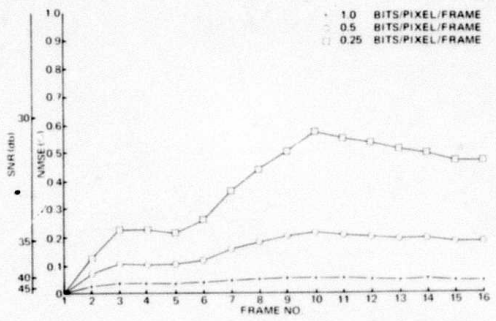


Fig. 2. Fourier/Fourier/DPCM coder at various bit rates.

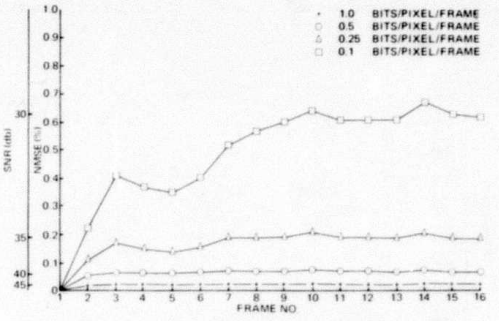
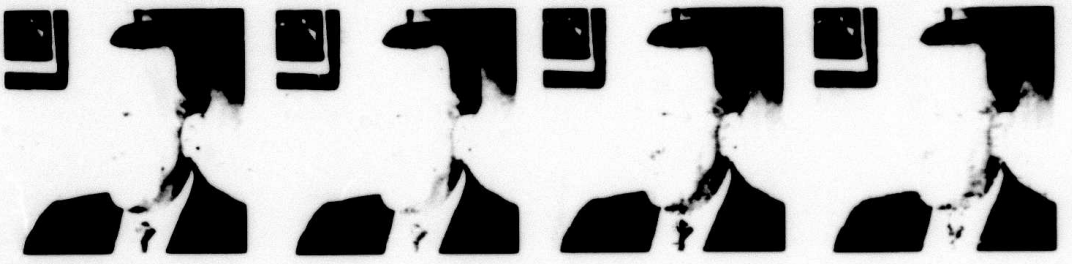


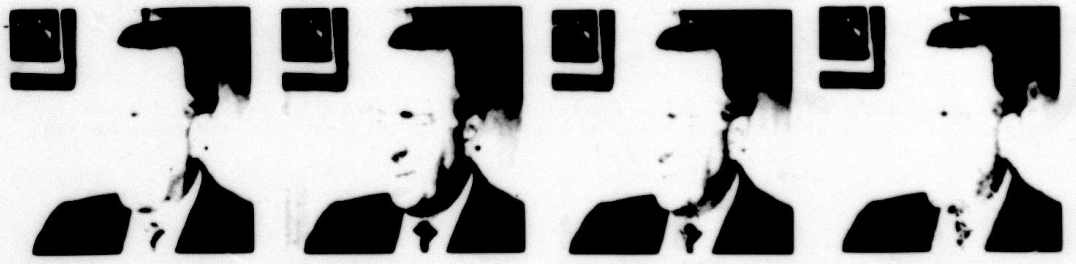
Fig. 3. Cosine/cosine/DPCM coder at various bit rates.

Photographs of frame number 16 after coding by the FFD and CCD coders at average pixel bit rates of 1.0, 0.5, 0.25, and 0.1 are shown as figures 4 and 5. The results shown in figure 4 for the FFD coder were obtained by coding the real and imaginary components of the Fourier coefficients by assigning half of the available bits to each component.



1.0 bits/pixel/frame 0.5 bits/pixel/frame 0.25 bits/pixel/frame 0.1 bits/pixel/frame

Fig. 4. FFD coder for frame 16.



1.0 bits/pixel/frame 0.5 bits/pixel/frame 0.25 bits/pixel/frame 0.1 bits/pixel/frame

Fig. 5. CCD coder for frame 16.

Noise Immunity

Performance of the FFD and CCD hybrid interframe coders was investigated in the presence of channel noise. In order to study the effect of channel noise, a binary symmetric channel was simulated. The channel is assumed to operate on each binary digit independently, changing each digit from 0 to 1 or from 1 to 0 with probability P_{ce} and leaving the digit unchanged with probability $1-P_{ce}$. At the receiver, the encoded picture is reconstructed from the string of binary digits, including errors, transmitted across the channel.

Degradations due to channel noise probabilities, P_{ce}, of zero, 10^{-3} and 10^{-2} for the FFD and CCD coders at average bit rates of 1.0 and 0.25 bits/pixel/frame are shown in figures 6 through 9. The generally monotonically increasing character of these curves illustrates the fact that once an error has occurred, it tends to propagate in the temporal direction until corrected by a frame refresh.

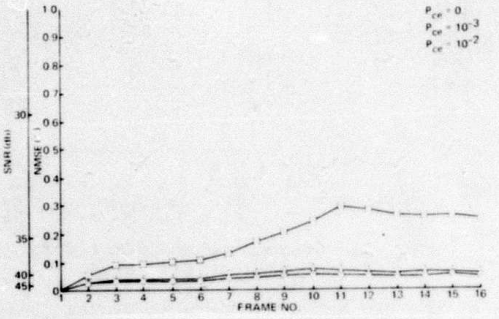


Fig. 6. Effects of channel noise for Fourier/Fourier/DPCM coder at an average 1.0 bits/pixel/frame.

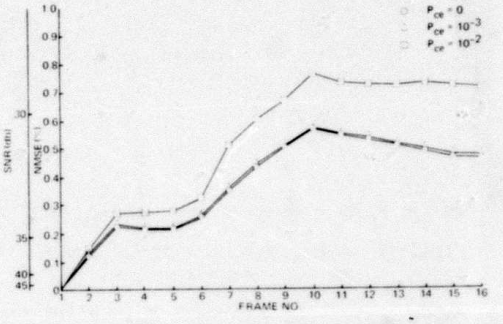


Fig. 7. Effects of channel noise for Fourier/Fourier/DPCM coder at an average 0.25 bits/pixel/frame.

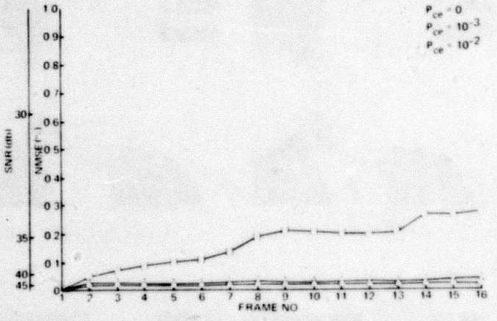


Fig. 8. Effects of channel noise for cosine/cosine/DPCM coder at an average 1.0 bits/pixel/frame.

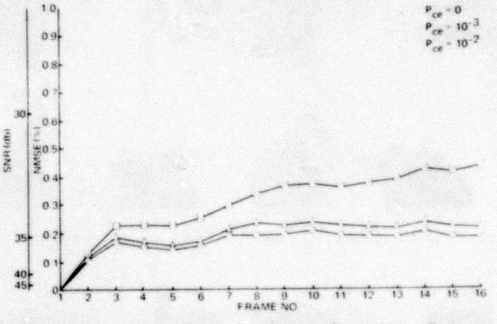
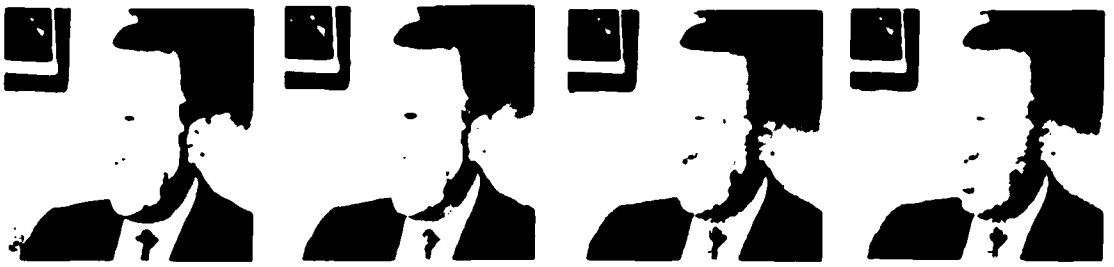


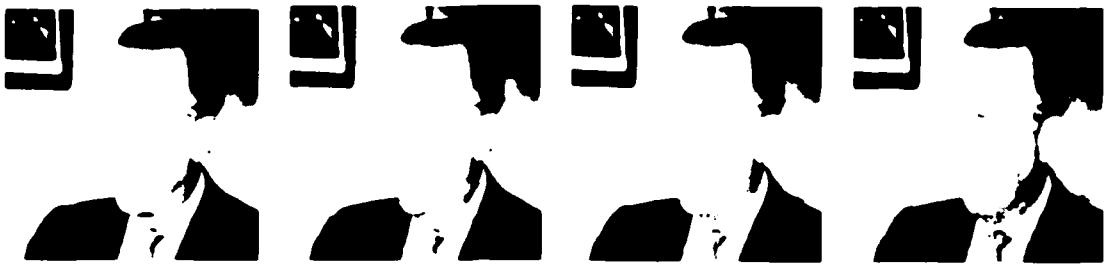
Fig. 9. Effects of channel noise for cosine/cosine/DPCM coder at an average 0.25 bits/pixel/frame.

Photographs corresponding to average bit rates of 1.0 and 0.25 bits/pixel/frame for the FFD and CCD coders with channel error probabilities of 10^{-3} and 10^{-2} are shown in figures 10 and 11.



1.0 bits/pixel/frame $P_{ce} = 10^{-3}$ | 1.0 bits/pixel/frame $P_{ce} = 10^{-2}$ | 0.25 bits/pixel/frame $P_{ce} = 10^{-3}$ | 0.25 bits/pixel/frame $P_{ce} = 10^{-2}$

Fig. 10. PFD coder with channel noise.



1.0 bits/pixel/frame $P_{ce} = 10^{-3}$ | 1.0 bits/pixel/frame $P_{ce} = 10^{-2}$ | 0.25 bits/pixel/frame $P_{ce} = 10^{-3}$ | 0.25 bits/pixel/frame $P_{ce} = 10^{-2}$

Fig. 11. CCD coder with channel noise.

Bit Transfer Rate

In keeping with the previously mentioned objective of minimizing the number of bits transmitted while retaining image fidelity, a series of experiments was performed in which certain bit transfer rates (BTR) across the channel were fixed. The BTR is defined as the product of average pixel bit rate per frame and frame rate and has units of bits/pixel/sec.

$$\text{BTR} = (\text{bits/pixel/frame}) \times (\text{frames/sec}) \tag{8}$$

The available 16 frame test data base was extracted from a 24 fps motion picture sequence. By employing frame skipping techniques, temporal subsampling was used to simulate short 12, 8 and 6 fps sequences from the 16 frame test data base.

Average bit rates in the interval 0.083 to 1.333 bits/pixel/frame were used in conjunction with the four frame rates mentioned above to perform simulations with BTR values of 8, 6, 4 and 2 bits/pixel/sec. The results of these experiments are shown as figures 12 through 15, respectively. For all cases examined, the graphs show that reduced frame rates produce smaller NMSE values for the individual frames coded. This indicates that reductions experienced in frame-to-frame correlations due to temporal subsampling are completely compensated for by the increased number of bits available for coding. However, subjectively, reduced frame rates tend to result in jerky subject motion. This is most apparent for rapidly moving objects in the field of view and is of lesser consequence for slowly changing scenes.

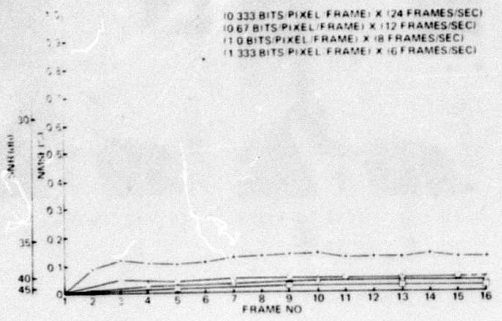


Fig. 12. Cosine/cosine/DPCM coder at bit transfer rate of 8 bits/ pixel/sec.

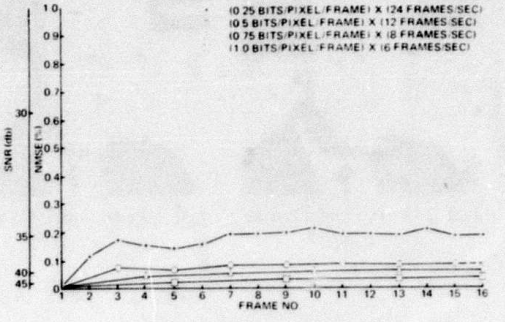


Fig. 13. Cosine/cosine/DPCM coder at bit transfer rate of 6 bits/ pixel/sec.

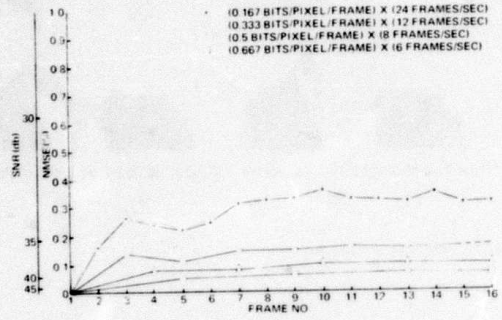


Fig. 14. Cosine/cosine/DPCM coder at bit transfer rate of 4 bits/ pixel/sec.

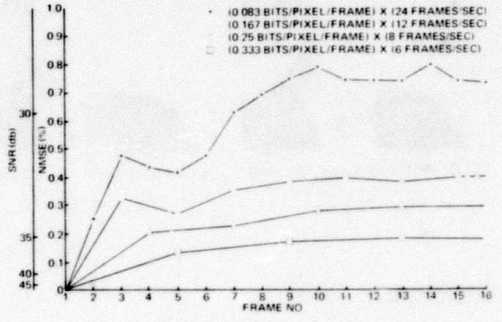


Fig. 15. Cosine/cosine/DPCM coder at bit transfer rate of 2 bits/ pixel/sec.

Conclusions

Based on theoretical and experimental results obtained to date, two main conclusions have been reached. The first is that exploitation of temporal correlations in addition to spatial correlations has been demonstrated to be a viable technique for coding sequences of digital images. This fact is demonstrated by a comparison of the average bit rates required for the interframe cosine/cosine/DPCM and the existing intraframe cosine/DPCM coders to achieve the same level of NMSE performance. The sixteenth frame of the test data base was chosen for comparison and was coded at an average 0.25 bits/pixel by the interframe cosine/cosine/DPCM coder. When using the intraframe cosine/DPCM coder, it was necessary to code this frame at a bit rate of more than 2 bits/pixel to achieve the same NMSE.

The second conclusion is that the performance of the hybrid interframe coders investigated are heavily data dependent. In the case of the 16 frame head and shoulders test data base, good coding performance was achieved since subject movement was restricted to a relatively small portion of the image. However, coding performance with a different aerial data base was degraded from the previous case due principally to platform motion which caused frame-to-frame pixel amplitude variations across the entire image. Since the performance of the hybrid interframe coders is dependent on temporal correlation, a reduced level of performance is to be anticipated for image sequences distorted by motion.

References

1. P. A. Wintz, "Transform Picture Coding," Proceedings IEEE, Vol. 60, No. 7, pp. 809-820, July 1972.

2. W. K. Pratt, L. R. Welch and W. Chen, "Slant Transforms for Image Coding," IEEE Transactions on Communications, Vol. COM-22, No. 8, pp. 1075-1093, August, 1974.

3. N. Ahmed, T. Natarajan, K. R. Rao, "Discrete Cosine Transform," IEEE Trans. on Computers, Vol. C-23, No. 1, pp. 90-93, January 1974.

4. W. K. Pratt, "Spatial Transform Coding of Color Images," IEEE Transactions on Communication Technology, Vol. COM-19, No. 6, pp. 980-992, December, 1971.

5. A. Habibi, "Comparison of n^{th} -order DPCM Encoder with Linear Transformation and Block Quantization Techniques," IEEE Transactions on Communication Technology, Vol. COM-19, No. 6, Part I, pp. 948-956, December 1971.

6. A. Habibi, "Hybrid Coding of Pictorial Data," IEEE Transactions on Communications, Vol. COM-22, No. 55, pp. 614-624, May 1974.

7. A. G. Tescher, "The Role of Phase in Adaptive Image Coding," Ph.D. Thesis, University of Southern California, Electrical Engineering Department, January 1974. Published as Report 510, University of Southern California, Image Processing Institute.

8. J. A. Roese, W. K. Pratt, G. S. Robinson and A. Habibi, "Interframe Transform Coding and Predictive Coding Methods," Proceedings of 1975 International Conference on Communications (ICC 75), Vol. II, Paper 23, pp. 17-21, June 16-18, 1975.

9. G. S. Robinson, "Orthogonal Transform Feasibility Study," NASA Final Report NASA-CR-115314, N72-13143 (176 pp.) (submitted by COMSAT Labs. to NASA Manned Spacecraft Center, Houston, Texas) November 1971.

10. A. J. Seyler, "Probability distributions of television frame differences," Proc. IREE, Australia, pp. 355-366, November 1965.

11. B. Smith, "Instantaneous companding of quantized signals," Bell System Technical Journal, Vol. 36, pp. 653-709, May 1957.

Acknowledgement

This research was supported by the Advanced Research Projects Agency of the Department of Defense and monitored by the Air Force Eastern Test Range under Contract R08606-72-0008 and by the Naval Undersea Center, San Diego, California, under Contract N00123-75-C-1192.

APPENDIX C

INTERFRAME TRANSFORM CODING AND PREDICTIVE CODING METHODS

INTERFRAME TRANSFORM CODING AND PREDICTIVE CODING METHODS *

John A. Roese
Naval Undersea Center
San Diego, California 92132

Ali Habibi
TRW Systems Group
Redondo Beach, California 90278

William K. Pratt Guner S. Robinson
Image Processing Institute
University of Southern California
Los Angeles, California 90007

ABSTRACT

Transform, differential pulse code modulation (DPCM), and hybrid transform/DPCM image coding methods are applied to coding successive frames of digital images. These coding techniques are designed to exploit the inherent spatial and temporal correlations of image sequences. In transform image coding, subsections of images are transformed into arrays of nearly uncorrelated coefficients by the use of orthogonal transformations. Bit compression is achieved by assigning the available bits in proportion to the energy of the transform coefficients. In DPCM image coding systems, the value of the image sample is predicted and the difference between the actual and the predicted value is quantized and transmitted. Hybrid transform/DPCM coding implementations combine the relatively superior low bit rate performance and channel noise immunity of transform methods with the minimal storage requirements of DPCM encoders. Interframe coding implementations studied include: three-dimensional Cosine transforms, hybrid two-dimensional spatial transforms with DPCM in the temporal direction, e.g., Cosine-Cosine/DPCM and Fourier-Fourier/DPCM, and three-dimensional DPCM. Results are summarized by evaluating the various implementations in terms of performance and complexity criteria.

INTRODUCTION

During the past few years several intraframe digital image coding systems based upon transform and linear predictive coding concepts have been developed [1, 2]. Most of these systems have achieved bit rate reductions by removal of statistical redundancies within an image frame combined with the deletion of that part of the spatial image representation least critical to the human observer. It is known that there is considerable temporal redundancy between frames of real time imagery systems; also there are psychophysical limitations to temporal image perception. Exploitation of either property in the past has been difficult to achieve because of implementation problems. However, there have been recent technological advances in signal processing circuitry which hold promise for the practical implementation of digital real time television image coders. Several such systems are described in this paper along with an analysis of their performance.

INTRAFRAME TRANSFORM AND PREDICTIVE CODING

Transform coding and linear predictive coding are, perhaps, the most widely employed techniques for intraframe image coding. Operational descriptions of these coding methods are given below.

Transform Coding: In transform coding systems, a one or two-dimensional mathematical transform of an image line segment or block is performed. The resulting transform coefficients are then quantized and coded. A bit rate reduction is possible because the distribution of energy in the transform domain permits more efficient quantization and coding. After decoding, an inverse transform is taken to obtain a replica of the image at the receiver. Transforms that have proven useful include the Fourier, Hadamard, Slant, Karhunen-Loeve, and Cosine transforms [3-5]. Transform coding techniques have been explored extensively both theoretically and by simulation. It has been shown that a significant bit rate reduction can be achieved in many applications with minimal image degradation. Simulation results indicate that a bit rate reduction to 1.5 bits/pixel can be obtained for monochrome image transform coding in 16 x 16 pixel blocks, while color images require about 2.0 bits/pixel [6]. The bit rate can be reduced further by making the transform coding system adaptive.

Predictive Coding: The high degree of correlation between a given pixel value and its nearest neighbors allows an image to be efficiently represented by coding only the difference between each pixel value and its predicted value. The predicted pixel value is based on previously scanned pixel values. In a differential pulse code modulation (DPCM) system, the predicted value of each pixel is subtracted from its actual value and this difference is quantized and transmitted. At the receiver, the quantized difference signal is combined with its predicted value to form the reconstructed pixel value. Basic DPCM image coding systems provide good quality at about 3 bits/pixel. Adaptive systems, in which parameters of the quantizer and predictor adapt to the image content, require about 2 bits/pixel.

INTERFRAME TRANSFORM CODING

In interframe transform coding, a three-dimensional unitary transform is performed on the data. Let $f(x, y, t)$ denote a three-dimensional array of amplitude values for each frame of a digital image. Also, let $F(u, v, w)$ be the three-dimensional array obtained by taking the three-dimensional transform in the (x, y, t) domain. If the size of the three-dimensional array is $N_1 x N_2 x N_3$, then such a transform coder can be described in a general form as

*To be published in ICC Conference Proceedings (Jan. 1975).

$$F(u,v,w) = \sum_{t=0}^{N_3-1} \sum_{y=0}^{N_2-1} \sum_{x=0}^{N_1-1} f(x,y,t)\emptyset(u,v,w,x,y,t) \tag{1}$$

$$f(x,y,t) = \sum_{w=0}^{N_3-1} \sum_{v=0}^{N_2-1} \sum_{u=0}^{N_1-1} F(u,v,w)\emptyset^{-1}(u,v,w,x,y,t)$$

where x and y are spatial coordinates, t is the time coordinate, u, v, w are the transform domain coordinates, and $\emptyset(u,v,w,x,y,t)$ represents a set of three-dimensional basis vectors. For the three-dimensional discrete Fourier transform, eq. (1) has the form,

$$F(u,v,w) = \frac{1}{N_1N_2N_3} \sum_{t=0}^{N_3-1} \sum_{y=0}^{N_2-1} \sum_{x=0}^{N_1-1} f(x,y,t) \cdot \exp\left[-2\pi i\left(\frac{ux}{N_1}+\frac{vy}{N_2}+\frac{wt}{N_3}\right)\right]$$

$$f(x,y,t) = \sum_{w=0}^{N_3-1} \sum_{v=0}^{N_2-1} \sum_{u=0}^{N_1-1} F(u,v,w) \cdot \exp\left[+2\pi i\left(\frac{ux}{N_1}+\frac{vy}{N_2}+\frac{wt}{N_3}\right)\right] \tag{2}$$

Since the Fourier kernel

$$\exp\left[\pm 2\pi i\left(\frac{ux}{N_1}+\frac{vy}{N_2}+\frac{wz}{N_3}\right)\right]$$

is separable, the three-dimensional transform can be computed as a one-dimensional transform in the temporal direction followed by a two-dimensional transform in the spatial domain. The two-dimensional spatial transform can, in turn, be computed as a one-dimensional transform along the rows followed by a one-dimensional transform along the columns. Thus, the basic one-dimensional transform that must be performed is

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \exp\left(-\frac{2\pi i}{N} ux\right) \tag{3}$$

for u=0, 1, . . . , N-1. In the case of the discrete Cosine transform [7], the one-dimensional transform is

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \cos\frac{(2x+1)u\pi}{2N} \tag{4}$$

for u= 0, 1, . . . , N-1.

The form of eq. (4) differs from that of reference [7] only by a normalization constant. An example of three-dimensional transform coding of a 16 frame data base is given in figure 1. This figure illustrates the decoded images for frames number 1 and 16. In this example, a three-dimensional Cosine transform was performed on cubic blocks of size 16x16x16 on 16 frames of size 256x256. The average bit rate used was 1.0 bits/pixel.

Frame 1 Frame 16

Figure 1

INTERFRAME DPCM CODING

In DPCM systems, linear prediction is used to generate a differential signal which is quantized and transmitted. At the receiver a similar predictor uses some previously transmitted values of the quantized differential signal to obtain a facsimile of the transmitted signal.

Prediction of a picture element is performed by using a set of previously scanned picture elements

$$\hat{s}_o = \sum_{i=1}^{n} a_i s_i \tag{5}$$

where $\{s_i\}$ is the set of picture elements with zero mean and variance σ^2, and n is the order of the predictor. The predictor parameters a_i are the solutions of n algebraic equations:

$$R_{oi} = \sum_{i=1}^{n} a_j R_{ij} \quad i=1,2,\ldots,n. \tag{6}$$

where R_{ij} are the correlations between picture elements:

$$R_{ij} = E\{s_i s_j\} \tag{7}$$

The variance of the differential signal

$$e = s_o - \hat{s}_o$$

is related to the variance σ^2 of picture elements by

$$\sigma_e^2 = \sigma^2 - \sum_{i=1}^{n} a_i R_{oi} \qquad (8)$$

In a three-dimensional DPCM interframe coder, the value of the picture element s_o is estimated using previously scanned picture elements as shown in figure 2 in which s_o is the picture element to be predicted. The elements s_1, s_2, s_4, and s_5 are the previously scanned picture elements in the present frame, and $s_3, s_6, s_7, \ldots, s_{12}, s_{13}$ are the closest samples on the previous frame.

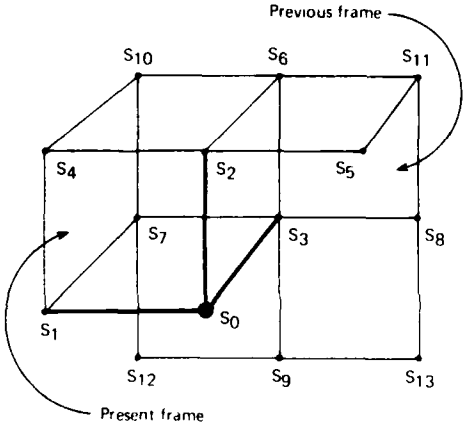


Figure 2

Figure 3 shows frames number 2 and 16 of the 16 frame data base after coding by a three-dimensional third order predictive coder at 2.0 bits/pixel. The picture elements used for predicting s_o are s_1 and s_2 on the present frame and s_3 on the previous frame.

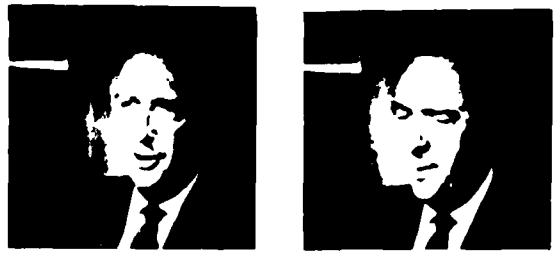
Frame 2 Frame 16

Figure 3

INTERFRAME HYBRID CODING

Three-dimensional hybrid encoders have been investigated which use two-dimensional transformations in the spatial domain cascaded with a DPCM encoder in the temporal domain. Due to DPCM encoding in the temporal direction, these encoders require storage of only the single previous frame of data. Simulation results indicate that the hybrid three-dimensional encoders perform better than the corresponding three-dimensional transform encoders. This observation is confirmed by the comparative performance of the two-dimensional hybrid and transform encoders previously reported in the literature [8]. The following discusses two different three-dimensional hybrid coding algorithms which have been investigated.

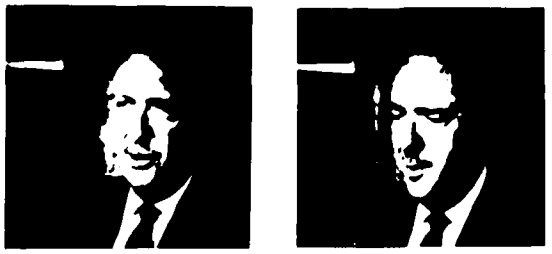
Two-Dimensional Cosine Transform/DPCM Coder: This hybrid coder exploits spatial image correlations by taking a two-dimensional discrete Cosine transform and temporal correlation by use of a DPCM coder. Theoretical studies indicate that this hybrid interframe coder possesses the attractive features of the hybrid intraframe coder. It is anticipated that this system will reduce the number of bits needed for reconstruction of television images at the receiver by a factor of five over the two-dimensional hybrid coder. Figure 4 shows the results for frames 1 and 16 after applying the hybrid Cosine-Cosine/DPCM encoder. The average bit rate used was 1.0 bits/pixel.



Frame 1 Frame 16

Figure 4

Two-Dimensional Fourier Transform/DPCM Coder: An alternative approach to the three-dimensional hybrid coder uses the two-dimensional Fourier transform in place of the two-dimensional Cosine transform. In this system, the two-dimensional Fourier transform coefficients of each frame are coded and transmitted. Figure 5 illustrates the results for frames 1 and 16 due to Fourier-Fourier/DPCM coding of the 16 frame data base. These results were obtained by coding the real and imaginary components of the Fourier coefficients at an average bit rate of 0.5 bits for each component.



Frame 1 Frame 16

Figure 5

The two-dimensional Fourier transform of the kth frame, $F_k(u, v)$, can be expressed as

$$F_k(u, v) = A_k(u, v)e^{j\theta_k(u, v)} \quad (4)$$

where $A_k(u, v)$ and $\theta_k(u, v)$ are the amplitude and phase planes of the kth frame. Many types of motions such as panned motion produces significant changes in the phase plane and small changes in the amplitude plane. An attractive feature of amplitude-phase coding is that motion compensation may be implemented directly in the phase plane by application of appropriate phase correction factors.

SYSTEMS ANALYSIS

In this section, direct comparisons are made between the four previously discussed interframe encoders. The comparison criteria used are: coding efficiency, storage requirements, noise immunity and implementation complexity.

The results shown as figures 1, 3, 4, and 5 illustrate the coding performance of the four encoders. For MSE comparison purposes, the encoders were run at an average bit rate of 1.0 bits/pixel. This bit rate was chosen as it is the lowest bit rate at which the three-dimensional DPCM encoder can operate. Additional experiments have shown that the hybrid transform/DPCM encoders can be successfully operated at even lower bit rates. For each encoder, MSE calculations were made for each frame coded. The MSE values were normalized relative to total image energy of each frame. Comparison of the normalized MSE values indicate that the hybrid Cosine-Cosine/DPCM and Fourier-Fourier/DPCM encoder implementations were superior in terms of MSE coding efficiency. This conclusion is supported by subjective comparison of the coded frames illustrated above.

A significant disadvantage of all interframe coding systems is the requirement for storage of previously scanned data frames. Of the interframe systems considered in this paper, the three-dimensional Cosine implementation is the least attractive in terms of required storage as several previous frames must be retained. Even if the number of frames stored is constrained to a low number, such as four, the total memory requirements are still excessive. This is evident when the three-dimensional Cosine encoder is compared to the three other implementations which use first-order predictive DPCM coding in the temporal domain. Under the assumption of first-order temporal predictors, only the single previous frame needs to be stored. A special case occurs for the three-dimensional DPCM implementation where, in addition to the previous frame, it is also necessary to retain the previously scanned line of the current frame.

Immunity to channel noise varies widely for the interframe coder implementations considered. The least sensitive is the three-dimensional transform encoder, whereas the DPCM encoder is most vunerable to channel noise due to its transmission of simple pixel amplitude differences at a low bit rate. The hybrid transform/predictive encoders transmit differences in transform coefficients instead of pixel amplitudes and are less sensitive to channel noise than strictly predictive encoders. An aspect of predictive interframe coders which has yet to be fully investigated is their amenability to use with error detection and correction algorithms.

The implementation complexity criterion is a coarse measure of weight, size and power requirements for each encoder. The inherently simple operation of DPCM coders combined with essentially a single frame of storage favors the DPCM interframe coder. Conversely, the multiple frame storage requirements of the three-dimensional transform coders severely limit their usefulness. The two hybrid transform/predictive encoders appear to be the best compromise as they combine single frame storage requirements with the simplicity of DPCM in the temporal domain.

A summary of the results of the systems analysis for the interframe encoders is contained in Table 1.

Encoder / Criteria	Three-dimensional Cosine	Hybrid Cosine-Cosine/DPCM	Hybrid Fourier-Fourier/DPCM	Three-dimensional DPCM
Coding efficiency* (normalized MSE)	0.163	0.017	0.025	0.168
Storage requirements (Number of frames stored)	N frames	2 frames	2 frames	2 frames
Complexity (size, weight and power)	High	Moderate	Moderate	Simple
Noise immunity	Very good	Good	Good	Poor

*For 4 frame, N=16 pixel, average bit rate of 1.0 bits/pixel

Table 1

REFERENCES

(1) W. K. Pratt, "Bibliography on Digital Image Processing and Related Topics," University of Southern California, USCEE Report 455, September 1, 1973.

(2) A. Habibi and G. S. Robinson, "A Survey of Digital Picture Coding," IEEE Computer, Vol. 7, No. 5, pp. 22-34, May, 1974.

(3) W. K. Pratt, J. Kane and H. C. Andrews, "Hadamard Transform Image Coding," Proc. IEEE, Vol. 57, pp. 58-68, January, 1969.

(4) W. K. Pratt, L. R. Welch and W. Chen, "Slant Transforms for Image Coding," IEEE Transactions on Communications, Vol. COM-22, No. 8, pp. 1075-1093, August, 1974.

(5) A. Habibi, W. K. Pratt, G. S. Robinson, R. Means, H. Whitehouse and J. Speiser, "Real Time Image Redundancy Reduction Using Tranform Coding Techniques," Proceedings of the 1974 International Communications Conference (ICC 74), Minneapolis, Minnesota, June 17-19, 1974, Paper (18A).

(6) W. K. Pratt, "Spatial Transform Coding of Color Images," IEEE Transactions on Communication Technology, Vol. COM-19, No. 6, pp. 980-992, December, 1971.

(7) N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete Cosine Transform," IEEE Transactions on Computers, Vol. C-23, No. 1, pp. 90-93.

(8) A. Habibi, "Hybrid Coding of Pictorial Data," IEEE Transactions on Communications, Vol. COM-22, No. 5, pp. 614-624, May, 1974.

ACKNOWLEDGEMENT

This research was supported by the Advanced Research Projects Agency of the Department of Defense and monitored by the Air Force Eastern Test Range under Contract RO8606-72-0008 and by the Naval Undersea Center, San Diego, California, under Contract N00123-73-C-1507.

APPENDIX D

THE PRIME COSINE TRANSFORM

THE PRIME COSINE TRANSFORM

J. Speiser

NAVAL UNDERSEA CENTER
SAN DIEGO, CA 92132

This note discusses a high speed implementation of the odd discrete cosine transform (ODCT) which eliminates the multipliers required in earlier implementations [1] based on the chirp-Z transform. The discrete cosine transform is useful for television data compression since its basis vectors closely approximate those of the optimum Karhunen-Loeve transform for exponentially correlated data [2].

The ODCT is defined as the first N Fourier coefficients of the length 2N-1 even extension of the data, assuming that the data consists of N real values. This is shown in eqn (1).

$$G_k = \sum_{n=-(N-1)}^{N-1} g_n \, e^{-i2\pi kn/(2N-1)} \quad \text{for } k=0,1,\ldots N-1 \qquad (1)$$

where $g_{-n} = g_n$

In order to be able to use a variant of Rader's Prime Transform algorithm [3] we assume that P = 2N-1 is a prime. Note that the data block length, N, need not be a prime, as shown in the Table.

It has been previously shown that if P is a prime, the discrete Fourier transform (DCT) of length P can be implemented using a circular convolution of length P-1 together with two analog permuter memories of length P [4]. It will be shown here that the symmetry of the extended data in equation (1) permits the size of the circular convolution and the permuter memories to be reduced by a factor of two.

For each prime P, there is an integer R, called a primitive root of P, such that the residues of R, R^2, ... R^{P-1} are all distinct modulo P, and include every nonzero residue modulo P [5]. Therefore, for each integer n not congruent to zero mod P, n can be represented uniquely as a power of R modulo P, say $n = R^n$ (mod P). The integer n' is called the index of n (mod P) with respect to the primitive root R. In effect, R plays the role of the base of a system of logarithms in modulo P arithmetic, and n' is the logarithm of n. This representation is useful because it allows us to replace multiplication by addition in the exponent of the DFT, and thus reduce the DFT to a circular correlation, as shown in equation (2).

$$G_{Rk'} = g_0 + \sum_{\substack{n'=-(N-1) \\ n' \neq 0}}^{N-1} g_{Rn'} \, e^{\frac{-i2\pi R^{n'+k'}}{2N-1}} \qquad (2)$$

Since zero does not have an index (logarithm) with respect to the primitive root, the zero frequency point in the transform must be computed separately, as shown in equation (3).

$$G_0 = \sum_{n=-(N-1)}^{N-1} g_n = g_0 + 2 \sum_{n=1}^{N-1} g_n \qquad (3)$$

The interpretation of equation (2) is that the DFT coefficients in permuted order are obtained by adding g_0 to the circular crosscorrelation of a permuted sinusoid with a permuted version of the data points excluding g_0. For the special case of the DCT, the symmetry of the extended data allows us to replace the complex exponential by a cosine, as shown in equation (4).

$$G_{Rk'} = g_0 + \sum_{\substack{n'=-(N-1) \\ n' \neq 0}}^{N-1} g_{Rn'} \cos\left(2\pi R^{k'+n'}/(2N-1)\right) \qquad (4)$$

It will now be shown that the permuted data and the permuted cosine have periodicity N-1, so that the circular correlation of length P-1 = 2N-2 can be reduced to a circular correlation of length N-1. First, note that both the extended data and the cosine function are even. Let h_n be any even sequence. It will be shown that h_{R^s} has period N-1, where the subscript R^s is reduced modulo P = 2N-1.

It is well known in number theory that $R^{(P-1)/2} = -1 \pmod{P}$ [5]. In our case, (P-1)/2 = (2N-2)/2 = N-1. Therefore $h_{R^s + (N-1)} = h_{R^s R^{N-1}} = h_{-R^s} = h_{R^s}$

Using this periodicity property applied to equation (4) lets us write the ODCT in shorter form, as shown in equation (5).

$$G_{Rk'} = g_0 + 2 \sum_{n'=1}^{N-1} g_{Rn'} \cos\left(2\pi R^{n'+k'}/(2N-1)\right) \qquad (5A)$$

$$G_o = g_0 + \sum_{n=1}^{N-1} g_n \qquad (5B)$$

The circular correlation required for equation 5A may be implemented by any of the alternative methods for implementing a circular correlation. The most straightforward would be to use a transversal filter of length 2(N-1)-1 = 2N-3, with tap weights of $\cos(2\pi R^s/(2N-1))$, for s = N-1,...1,0,1,2...N-1.

The architecture of the transform is shown in Figure 1, and is virtually identical to that of the prime Fourier transform, except that the prime cosine transform need only permute real data and filter the permuted real data with a filter having real weights. The prime cosine transform is thus considerably simpler to implement than a prime Fourier transform of the same block length.

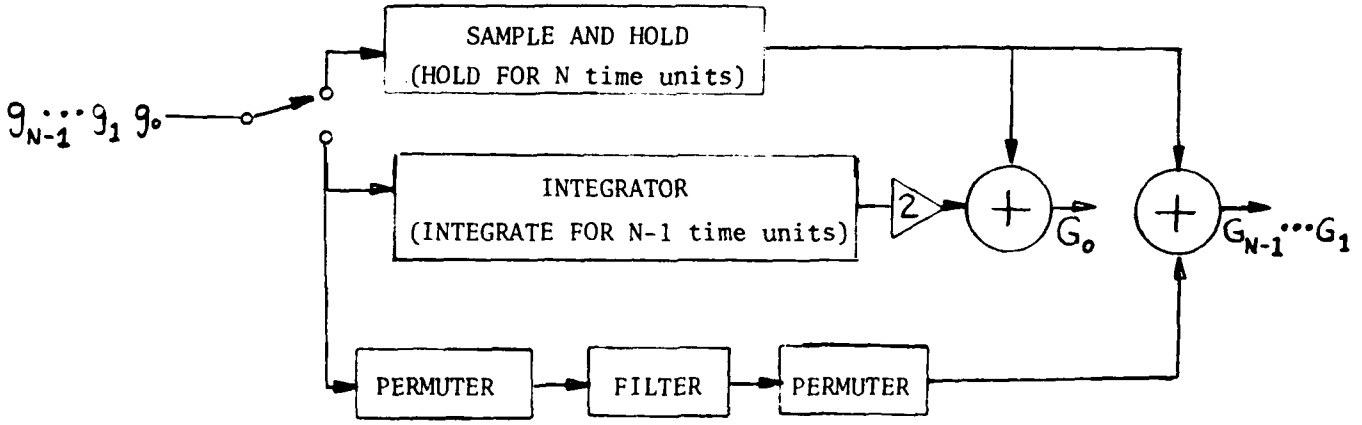
A suitable analog permuter memory has been developed for the Naval Undersea Center as a minor modification of a commercially available serial analog memory*. The commercially available serial access memory stores analog samples as charges in an array of MOS capacitors under the control of read in and read out shift registers. The permuter memory shown in Figure 2 differs only in the fact that one of the shift registers has been replaced by a binary decoder, thus allowing the data to be reordered by an external control signal.

*Reticon Corp. Sunnyvale, CA.

P (prime)	N = (P+1)/2 (data block length)	2N-3 = P-2 = filter length
31	16	29
61	31	59
127	64	125
251	126	249
257	129	255

Table 1

Selected Primes and the corresponding OCDT lengths and filter lengths. (The filter lengths shown assume that the data is not recirculated or reread into the filter.)

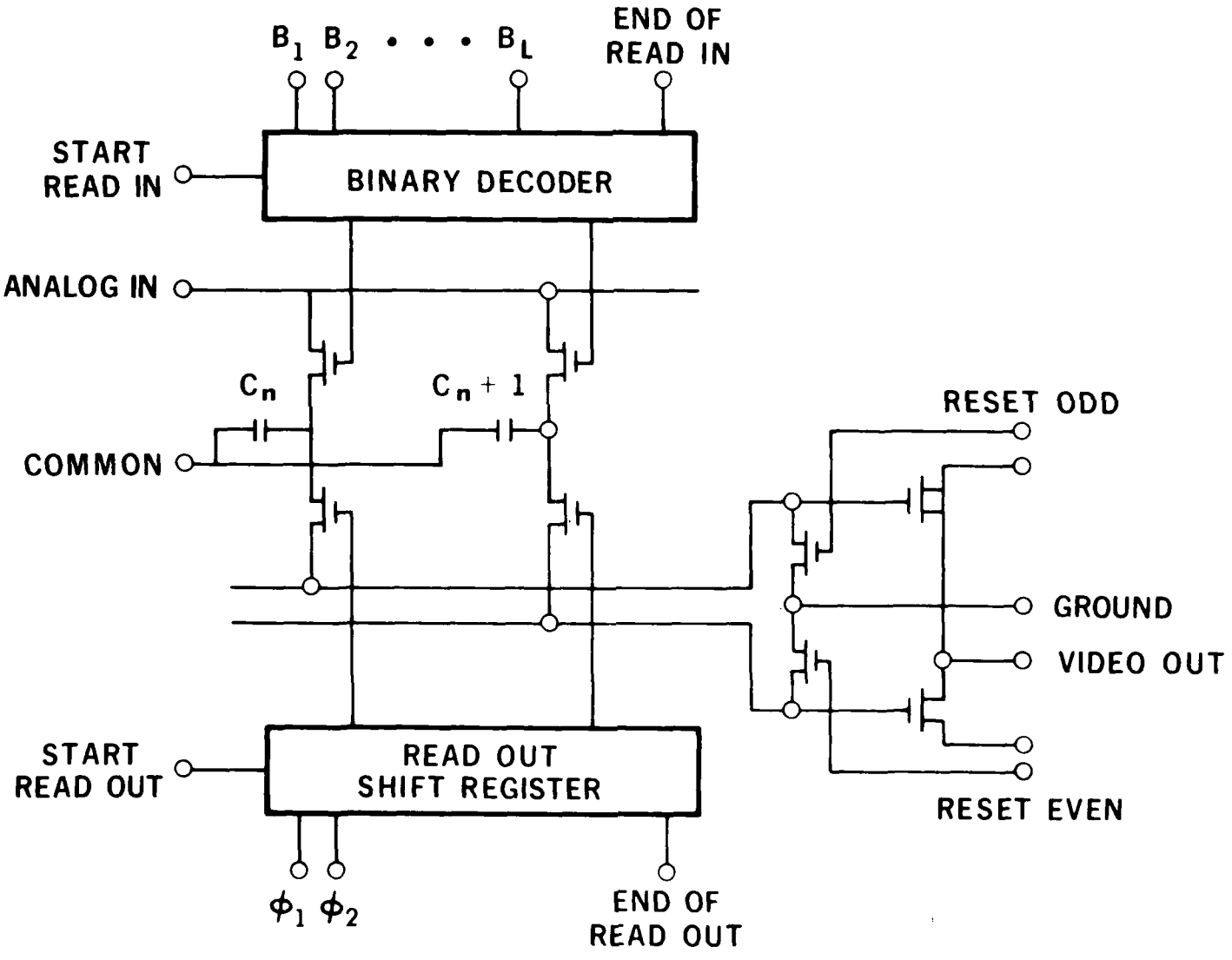


(the switch is in the up position for the first data sample, and is down for the remaining N-1 samples)

Figure 1. Prime ODCT architecture.

FIGURE 2

ANALOG PERMUTATION MEMORY



REFERENCES

1. Speiser, J. M., "High Speed Serial Access Implementations For Discrete Cosine Transforms," NUC TN 1265, 8 Jan 1974.

2. Means, R. W., H. J. Whitehouse, and J. M. Speiser, "Real Time TV Image Redundancy Reduction Using Transform Techniques," New Directions in Signal Processing in Communication and Control, Nato Advanced Study Institutes Series, Noordhoff, Leyden, 1975.

3. Rader, C., "Discrete Fourier Transforms When the Number of Data Samples is Prime," Proc. IEEE, Vol. 56, pp 1107-1108, 1968.

4. Whitehouse, H. J., R. W. Means, and J. M. Speiser, "Signal Processing Architectures Using Transversal Filter Technology," to appear in the Proceedings of the 1975 IEEE International Symposium on Circuits and Systems (Boston, April 21-23, 1975) paper SC.1.

5. Gauss, C. F., "Disquisitiones Arithmeticae, Lipsiae, 1801. Also available in English translation by Arthur A. Clarke, Yale University Press, New Haven, 1966, (see especially pp. 37-40). Although primitive roots are also discussed in most modern textbooks on number theory, Gauss' original exposition remains unrivalled for clarity.

APPENDIX E

PRIME TRANSFORM SAW DEVICE

PRIME TRANSFORM SAW DEVICE *

J. M. Alsup
Naval Undersea Center
San Diego, CA 92132

ABSTRACT. A method of calculating the discrete Fourier transform through the prime transform algorithm with surface acoustic wave devices is presented. The method is similar to the chirp-Z transform (CZT) technique [4], and utilizes the SAW device as a transversal filter. The prime transform is based on an algorithm which uses as indices the set of integers generated by the series R^k modulo N ($k = 1, 2, \ldots, N-1$). N is prime and R is an integer whose special property relative to N is that its successive powers modulo N are distinct, and therefore form a permutation of the integers $1, 2, \ldots, N-1$. The SAW prime transform implementation has the same processing speed advantage as the SAW CZT implementation, namely, that it computes a discrete Fourier transform with speed commensurate to a fully pipelined FFT running at the same sample rate. The attributes of small size, light weight, and interconnection simplicity are also maintained.

The Prime Transform

The discrete Fourier transform (DFT) of a sampled data sequence has special properties when the number of points to be transformed is prime [1]. For each prime number N there exist integers, known as "primitive roots," whose successive integer powers modulo N will generate a permuted version of the sequence $1, 2, \ldots, N-1$ [2]. The DFT (1) can then be written in terms of the permuted integer sequence for non-zero values of the time and frequency indices, so that (2) results.

$$G_m = \sum_{n=0}^{N-1} g_n \exp(-j2\pi mn/N) \qquad (1)$$

$$G_{R^k} = g_0 + \sum_{\ell=1}^{N-1} g_{R^\ell} \exp(-j2\pi R^{k+\ell}/N) \qquad (2)$$

where

$$m = R^k \bmod N,\ k = 0, 1, \ldots, N-2,$$

$$n = R^\ell \bmod N,\ \ell = 1, 2, \ldots, N-1,$$

and

$$G_o = \sum_{n=0}^{N-1} g_n .$$

Equation (2) represents a circular convolution with auxiliary operations, and as such is suitable for implementation by transversal filter architectures [3]. The filter tap weights are given by the expression $\exp(-j2\pi R^k/N)$, and are permuted values of a complex sinusoid. The sequence g_{R^ℓ} is a permuted version of the input sample values with the first sample deleted. The transform output coefficients G_{R^k} are the m = 1 to N - 1 Fourier coefficients in permuted order, and G_o is computed separately. A comparison of the basic prime transform implementation with that of the CZT is shown in Figure 1 (also see [4, 5]).

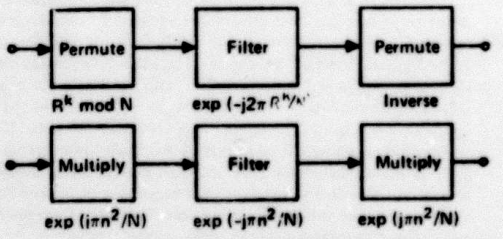


Figure 1. Block Diagram Comparison (simplified) between Prime Transform and CZT Transversal Filter Implementations

Direct permutation of sampled analog data can be achieved, for example, with a modified version of a commercially available device.[1] The auxiliary computation to account for the contribution of the zeroth sample can be carried out in a number of different ways, as are shown in Figure 2.

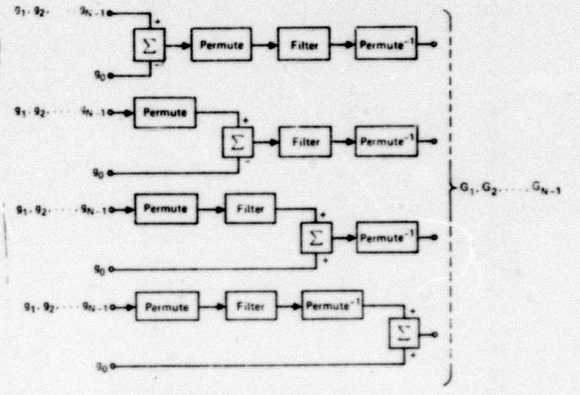


Figure 2. Four Ways to Include Auxiliary Computation for Zeroth-Sample Input.

For a given prime N, there exists a selection of integers between 0 and N which qualify as primitive roots. The number of such roots is given by the Euler ϕ-function, $\phi(N-1)$. For example, there are 8 primitive roots associated with N = 31: 3, 11, 12, 13, 17, 21, 22, and 24. Thus, there are $\phi(N-1)$ different ways to fabricate a prime transform device of length N. The variety of related pulse compression codes is discussed in [7]. Primitive roots for various primes are listed in [8]. Equation (3) illustrates the Euler ϕ-function:

$$\phi(A) = A \cdot \frac{a-1}{a} \cdot \frac{b-1}{b} \cdot \frac{c-1}{c} \cdots \qquad (3)$$

where $A = a^\alpha b^\beta c^\gamma \ldots$, and a, b, c, . . . are distinct primes, and $\alpha, \beta, \gamma, \ldots$ are integers.

[1] Reticon Corp. Model SAM (sampled analog memory).

*To be published in Ultrasonics Symposium Proceedings. IEEE (1975).

SAW Implementation

A surface acoustic wave (SAW) device was designed and constructed at NUC to verify the feasibility of the prime transform so implemented. The prime number was chosen to be 31, and the primitive root, 3. This led to the selection of the tap weights tabulated in Table I. Since circular convolution was required, two periods less one sample of the 30-tap sequence were incorporated into the tapping structure, resulting in a total of 59 complex taps. The complex arithmetic was implemented by using real and imaginary parts of the specified tap response to determine amplitude weightings on two parallel acoustic paths driven by a common acoustic input signal.

Table I. Prime Transform Tap Weights, N = 31, R = 3.

k	p = 3K mod 31	cos 2π p/N	-sin 2π p/N
1	3	.821	-.571
2	9	-.251	-.968
3	27	.689	.725
4	19	-.759	.651
5	26	.529	.849
6	16	-.995	.101
7	17	-.954	.299
8	20	-.612	.791
9	29	.919	.394
10	25	.347	.938
11	13	-.874	-.485
12	8	-.050	-.999
13	24	.151	.988
14	10	-.440	-.898
15	30	.980	.201
16	28	.821	.571
17	22	-.251	.968
18	4	.689	-.725
19	12	-.759	-.651
20	5	.529	-.849
21	15	-.995	-.101
22	14	-.954	-.299
23	11	-.612	-.791
24	2	.919	-.394
25	6	.347	-.938
26	18	-.874	.485
27	23	-.050	.999
28	7	.151	-.988
29	21	-.440	.898
30	1	.980	-.201

The entire electrode structure and a closup are shown in Figures 3 and 4. The device was built using AT-cut quartz as the substrate, with tap spacings of .0256 inch, equivalent to 0.2 μsec. Each tap consists of 8 finger-pairs with wavelength .0016 inch, equivalent to 80 MHz. The taps are spaced with a gap equal to their width, and the entire array of taps with one input transducer is 1.88 inches in length. The aperture of the taps varies according to the specified weighting function, and has a maximum of 0.256 inch. The aperture of the input transducer at either end of the device is 0.596 inch. The interconnect busses were configured so that either all 59 or just 30 taps (from either end) could be connected together to form the real and imaginary output signal components.

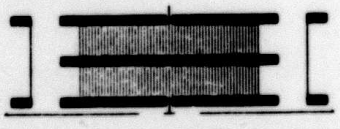


Figure 3. Prime Transform SAW Electrode Structure.

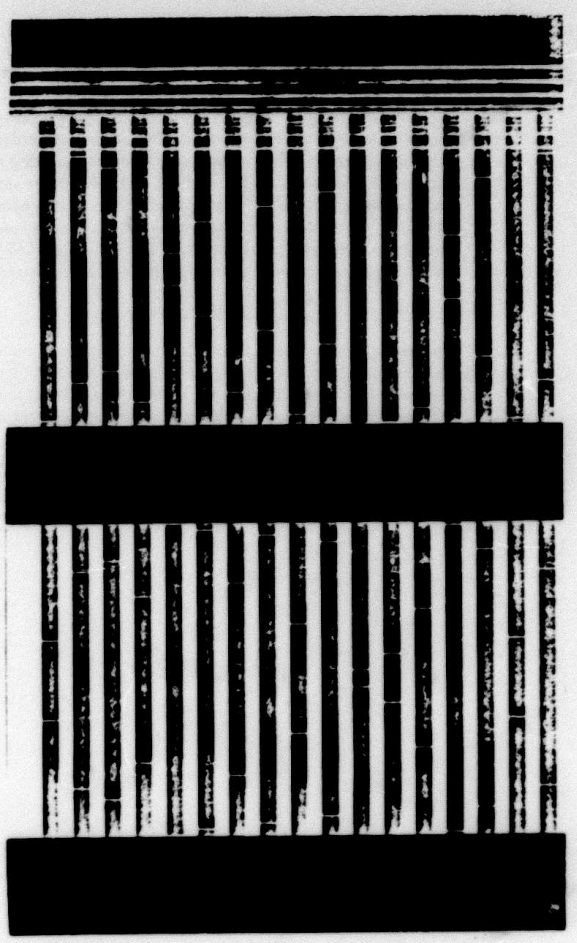


Figure 4. Prime Transform SAW Electrode Structure, Closeup.

A separate, uniformly weighted tapping structure (not shown) was designed to carry out the zeroth-sample auxiliary calculation, but was not used in initial device tests. The resulting device is capable of calculating a 31-point DFT in 6.2 μsec, and can operate at duty cycles up to 50%. Two such devices can be used alternately to achieve 100% duty cycle when required.

Experimental Results

The procedure used to view the impulse response of the SAW prime transform device is illustrated in Figure 5. A quarter-wave delay (about 3.1 ns) was used to enable quadrature combination of the real and imaginary parts of the response, and the separate and combined signals are shown in Figures 6 and 7. This technique for achieving discrete phase modulation of a signal carrier is discussed in [6], and could have just as easily been incorporated into the initial mask design instead of being left to electrical manipulation external to the SAW device as performed here. Lack of perfection in implementation of the specified tap weights due to missing or broken fingers will show up as a nonuniformity in the complex output sample magnitude, and Figure 7 does show this to some extent.

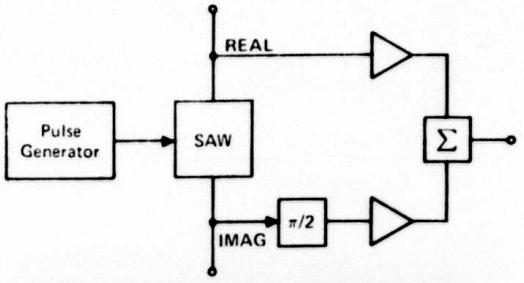


Figure 5. Experimental Setup, SAW Prime Transform Impulse Response.

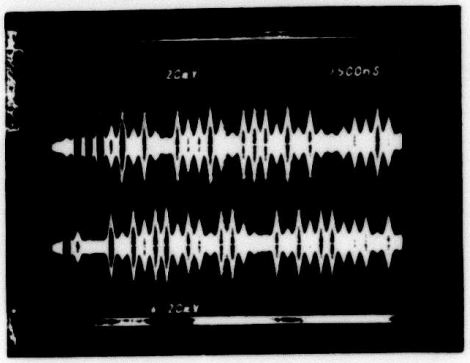


Figure 6. Impulse Response, Taps 30-1, SAW Prime Transform: Upper, [illegible] Part; Lower, [illegible] Part.

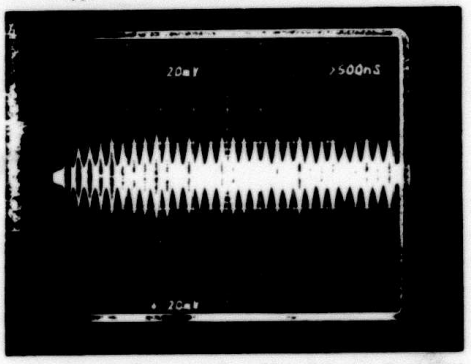


Figure 7a. Impulse Response, Taps 30-1, Complex Combination of Real and Imag Parts.

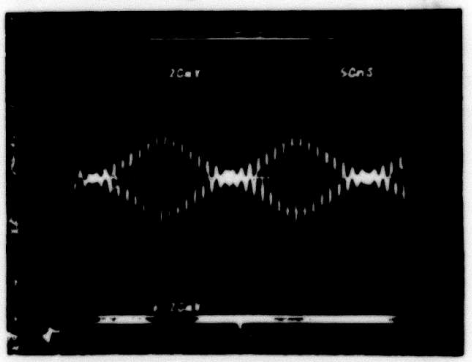


Figure [illegible] Complex Impulse Response, Closeup.

The 30 tap SAW transform device was evaluated in a similar configuration as shown in Figure 8. For diagnostic purposes, the 30-simple complex impulse response derived from one SAW [illegible] by the second SAW filter. The transform outputs for two diagnostic inputs are shown in Figures 9 and 10. These outputs are not corrected for zeroth-simple contribution, and so therefore should look like single peaks at the appropriate frequency bin, with uniform sidelobes down 30 dB at the other 29 frequency bins. Figure 9 shows this general appearance, with sidelobe level at about −15 dB. Figure 10 shows the same outputs after squaring via a double-balanced mixer, and low-pass filtering. The bell shape of the transform peak (see closeup, Figure 9) is mathematically the shape of the convolution of two isosceles triangles [9]. It is believed that performance close to theoretical can be achieved when defect-free devices are used.

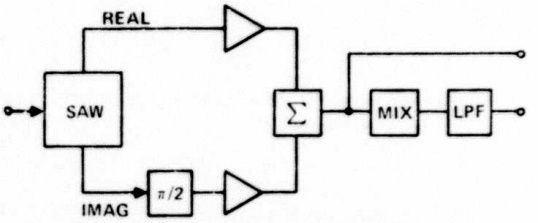


Figure 8. Experimental Setup, SAW Prime Transform Outputs.

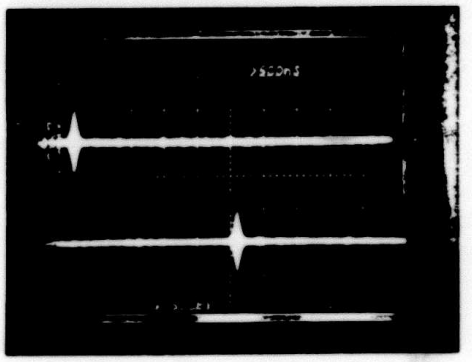


Figure 9a. SAW Prime Transform Outputs at Carrier (80 MHz): Upper, Input frequency = 1 cycle; Lower, Input Frequency = 30 cycles.

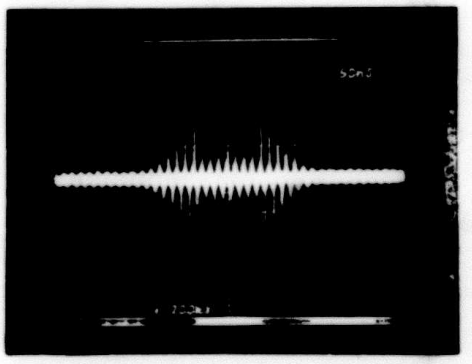


Figure 9b. SAW Prime Transform Output, Input Frequency = 30 cycles (closeup).

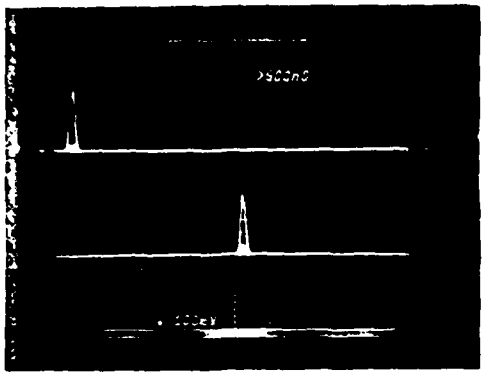


Figure 10. SAW Prime Transform Outputs, Magnitude Squared, at Baseband: Upper, Input Frequency = 1 cycle; Lower, Input Frequency = 30 cycles.

Conclusions

The prime transform algorithm implemented with SAW transversal filters has been shown to be a powerful tool for the high-speed evaluation of the discrete Fourier transform. At a 5 MHz sample rate, a 31-point DFT can be calculated in 6.2 μsec, which represents a substantial increase in speed over conventional FFT techniques. The transversal filter prime transform is very similar to the CZT method, and represents an alternative to it when permutation of the data being transformed is preferred over multiplication of the data. The prime transform requires auxiliary calculations to account for the contribution of the zeroth data sample and to evaluate the dc transform coefficient, but either or both of these may be dispensed with if the application does not depend critically upon them. Inverse permutation of the filter output may also be eliminated if not needed.

The same limitations with regard to accuracy apply to analog implementations of the prime transform algorithm such as this as also apply to CZT analog implementations, which limitation is about the equivalent of 7 or 8 bits. Like the CZT filter, the transversal filter prime transform implementation affords the possibility of calculating both forward and inverse transforms using identical modules, and this in turn leads to the capability of real-time high-data-rate linear signal processing tools such as cross-convolvers, cross-correlators, adaptive filters, and programmable matched filters.

References

[1] Rader, C., "Discrete Fourier Transforms When the Number of Data Samples is Prime," Proc. IEEE, Vol. 56, pp. 1107-1108, 1968.

[2] Gauss, C. F., Disquisitiones Arithmeticae, (translated by Arthur A. Clarke, Yale University Press, New Haven and London, 1966) Lipsiae, 1801.

[3] Whitehouse, H. J., R. W. Means, and J. M. Speiser, "Signal Processing Architectures Using Transversal Filter Technology," IEEE Advanced Solid-State Components for Signal Processing, IEEE International Symposium on Circuits and Systems, Newton, Mass., April 1975, pp. 5-29.

[4] Alsup, J. M., R. W. Means, and H. J. Whitehouse, "Real Time Discrete Fourier Transforms Using Surface Acoustic Wave Devices," Proc IEE International Specialist Seminar on Component Performance and Systems Application of Surface Acoustic Wave Devices, Aviemore, Scotland, September 24-28, 1973.

[5] Alsup, J. M., "Surface Acoustic Wave CZT Processors," Proceedings 1974 Ultrasonics Symposium, Milwaukee, Wis., Nov. 1974, pp. 378-81 (IEEE Cat. #74 CHO 896-1SU).

[6] Whitehouse, H. J., "Optical and Acoustical Signal Processing," Optical and Acoustical Micro-Electronics, New York, Polytechnic Press, Vol XXIII, April 1974, pp. 39-56.

[7] Alsup, J. M. and J. M. Speiser, "Exponential Residue Codes," to be published in Trans. Aerospace and Electronic Systems, Nov.-Dec., 1975.

[8] Osborn, R., Tables of All Primitive Roots of Odd Primes Less than 1000, University of Texas Press, Austin, 1961.

[9] Alsup, J. M., "Acoustic Surface Wave Golay-Coded Matched Filter," Naval Undersea Center TN 981, San Diego, CA, March 1973.

APPENDIX F

MODULAR IMPLEMENTATIONS OF THE DISCRETE COSINE TRANSFORM

J.M. Speiser
Naval Undersea Center

ABSTRACT

A modular architecture is described for the implementation of the Even Discrete Cosine Transform (EDCT). This architecture permits the use of four EDCT modules to compute a double length transform with twice the throughput rate of the individual modules.

INTRODUCTION

The utility of the EDCT for data compression has been previously described [1-3]. Several different serial access implementations for its high speed computation have also been described [4]. Implementations using charge coupled device (CCD) transversal filters are particularly attractive [2] for applications which require low weight, small size, low power consumption, and controllable clocking of the computation. Present CCD transversal filters perform well at shift rates of up to about 5×10^6 samples per second. This is a factor of two too slow to handle conventional television signals using the previously described EDCT architectures. This note describes a subdivision of the computation tasks to permit greater parallelism in the hardware to increase both the throughput and transform size implementable with a fixed set of transversal filter and chirp read-only memory modules.

DCT IMPLEMENTATIONS

A system to compute the EDCT of length N, implementing previously given equations [4] is shown in Fig. 1. It differs from serial access implementations of the ODCT [2] primarily in the sinusoidal multiplication following the chirp postmultiplication. Despite this slight complication the EDCT was selected for modular decomposition rather than the ODCT because an odd length extension of a data block can only be subdivided into a even length data block and an odd length data block, while an even length extension of a data block can be subdivided into two data blocks of the same length in order to permit simultaneous computation by similar modules.

Since intermediate complex quantities need to be preserved in the modular decomposition, the complex extended discrete Fourier transform (EDFT) portion of

Fig. 1 was chosen as the basic module. This module computes N points of a length 2N DFT of the extension of the data sequence by N zeroes. The interconnection of modules with minor auxilliary components to perform an EDCT with doubled throughput on a double length data block is shown in Fig. 2.

DERIVATION OF THE MODULAR EDCT

Let the input data be denoted by g_m for m= 0,1,...M-1. Define the symmetrized extension of the data by $g_{-1-m} = g_m$ for m=0,1,...M-1. The even discrete cosine transform (EDCT) of g may then be defined by any of the equivalent expressions shown in equations 1-3.

$$G_k = e^{\frac{-i\pi k}{2M}} \sum_{m=-M}^{M-1} g_m \, e^{\frac{-i2\pi km}{2M}} \qquad \text{for } k=0,1,\ldots M-1 \tag{1}$$

$$G_k = 2 \sum_{m=0}^{M-1} g_m \cos(\pi(m+.5)k/M) \qquad \text{for } k=0,1,\ldots M-1 \tag{2}$$

$$G_k = 2 \, \mathrm{Re} \left\{ e^{\frac{-i\pi k}{2M}} \sum_{m=0}^{M-1} g_m \, e^{\frac{-i2\pi km}{2M}} \right\} \qquad \text{for } k=0,1,\ldots M-1 \tag{3}$$

Let the data block length be even, say M=2N. Then the summation in the DCT can be split into shorter sums as shown in equations 4-5.

$$G_k = 2 \, \mathrm{Re} \left\{ e^{\frac{-i\pi k}{4N}} \sum_{m=0}^{2N-1} g_m \, e^{\frac{-i2\pi km}{4N}} \right\} =$$

$$2 \, \mathrm{Re} \left\{ e^{\frac{-i\pi k}{4N}} \left\{ \sum_{n=0}^{N-1} g_{2n} \, e^{\frac{-i2\pi k(2n)}{4N}} + \sum_{n=0}^{N-1} g_{2n+1} \, e^{\frac{-i2\pi k(2n+1)}{4N}} \right\} \right\} \qquad \text{for } k=0,1,\ldots 2N-1 \tag{4}$$

$$G_k = 2\,\mathrm{Re}\left\{ e^{\frac{-i\pi k}{4N}} \sum_{n=0}^{N-1} g_{2n}\, e^{\frac{-i2\pi kn}{2N}} \right\} + 2\,\mathrm{Re}\left\{ e^{\frac{-i3\pi k}{4N}} \sum_{n=0}^{N-1} g_{2n+1}\, e^{\frac{-i2\pi kn}{2N}} \right\}$$

$$\text{for } k=0,1,\ldots 2N-1 \qquad (5)$$

Each of the summations in equation 5 may be interpreted as a DFT of length 2N of a length N data block extended by N zeroes. The first N coefficients of such a DFT are computed directly by the EDFT module of Fig. 1. The remaining coefficients may all be written in the form G_{p+N}, where p=0, 1,...N-1. The corresponding terms of equation 5 are examined in equations 6-9.

$$e^{\frac{-i\pi(p+N)}{4N}} = e^{\frac{-i\pi}{4}}\, e^{\frac{-i\pi p}{4N}} \quad \text{for } p=0,1,\ldots N-1 \qquad (6)$$

$$e^{\frac{-i3\pi(p+N)}{4}} = e^{\frac{-i3\pi}{4}}\, e^{\frac{-i3\pi p}{4N}} \qquad (7)$$

$$\text{for } p=0,1,\ldots N-1$$

$$\sum_{n=0}^{N-1} g_{2n}\, e^{\frac{-i2\pi(p+N)n}{2N}} = \sum_{n=0}^{N-1} (-1)^n\, g_{2n}\, e^{\frac{-i2\pi pn}{2N}} \qquad (8)$$

$$\text{for } p=0,1,\ldots N-1$$

$$\sum_{n=0}^{N-1} g_{2n+1}\, e^{\frac{-i2\pi(p+N)n}{2N}} = \sum_{n=0}^{N-1} (-1)^n\, g_{2n+1}\, e^{\frac{-i2\pi pn}{2N}} \qquad (9)$$

$$\text{for } p=0,1,\ldots N-1$$

The structure shown in Figure 2 uses equation 4 to generate the coefficients $G_0, G_1, \ldots G_{N-1}$ and uses equations 6-9 to generate $G_N, G_{N+1}, \ldots G_{2N-1}$.

REFERENCES

1. Means, R. W., H. J. Whitehouse, et al, Image Transmission Via Spread Spectrum Techniques, ARPA Quarterly Technical Report, March 1 - June 1, 1973, Order Number 2303, Code Number 3G10.

2. Means, R. W., J. M. Speiser, H. J. Whitehouse, et al, Image Transmission Via Spread Spectrum Techniques, ARPA Quarterly Technical Report, June 1 - October 1, 1973, Order Number 2303, Code Number 3G10.

3. Ahmed, N., T. Natarajan, and K. R. Rao, On Image Processing and a Discrete Cosine Transform, IEEE Transactions on Electronic Computers, Jan. 1974, pp.90-93

4. Speiser, J. M., High Speed Serial Access Implementation For Discrete Cosine Transforms, NUC TN 1265, 8 January 1974

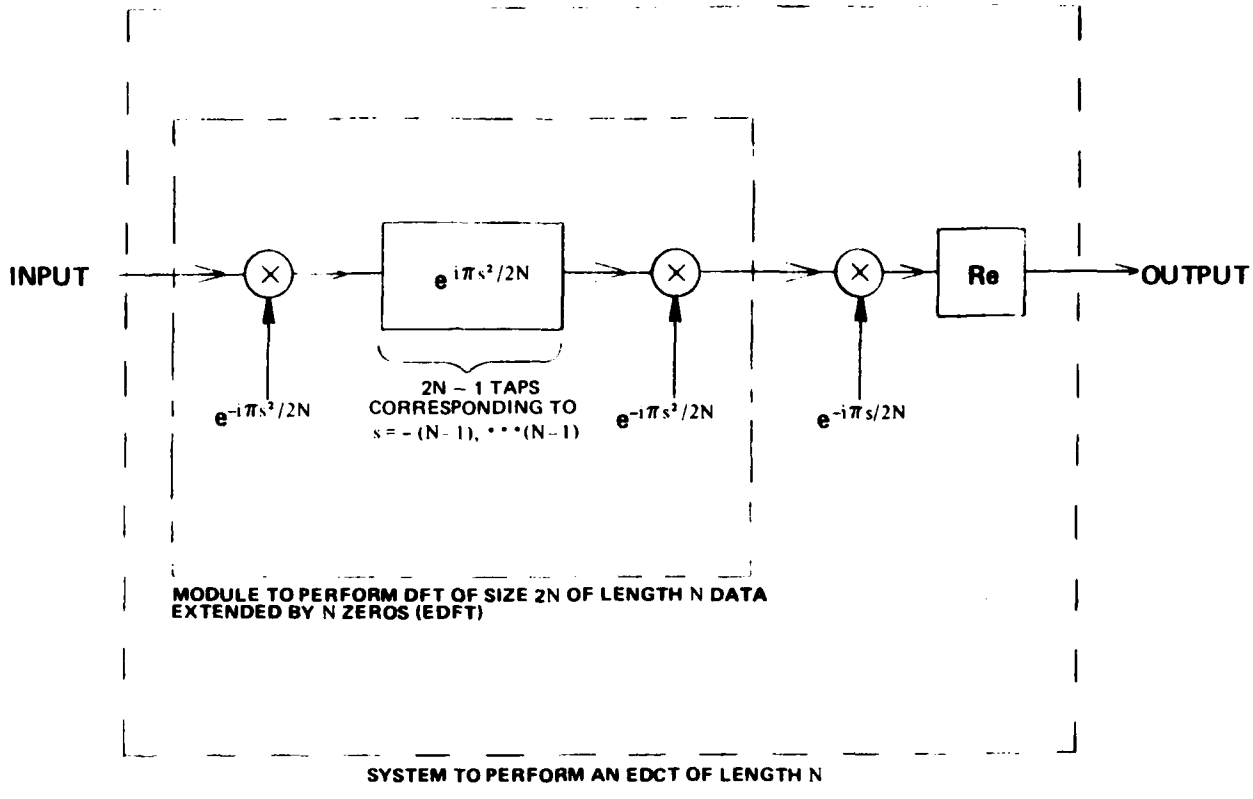


Fig. 1 System to Perform an EDCT of Length N Including Module to Perform a DFT of Size $2N$ of a Data Block of Size N Extended by N Zeros.

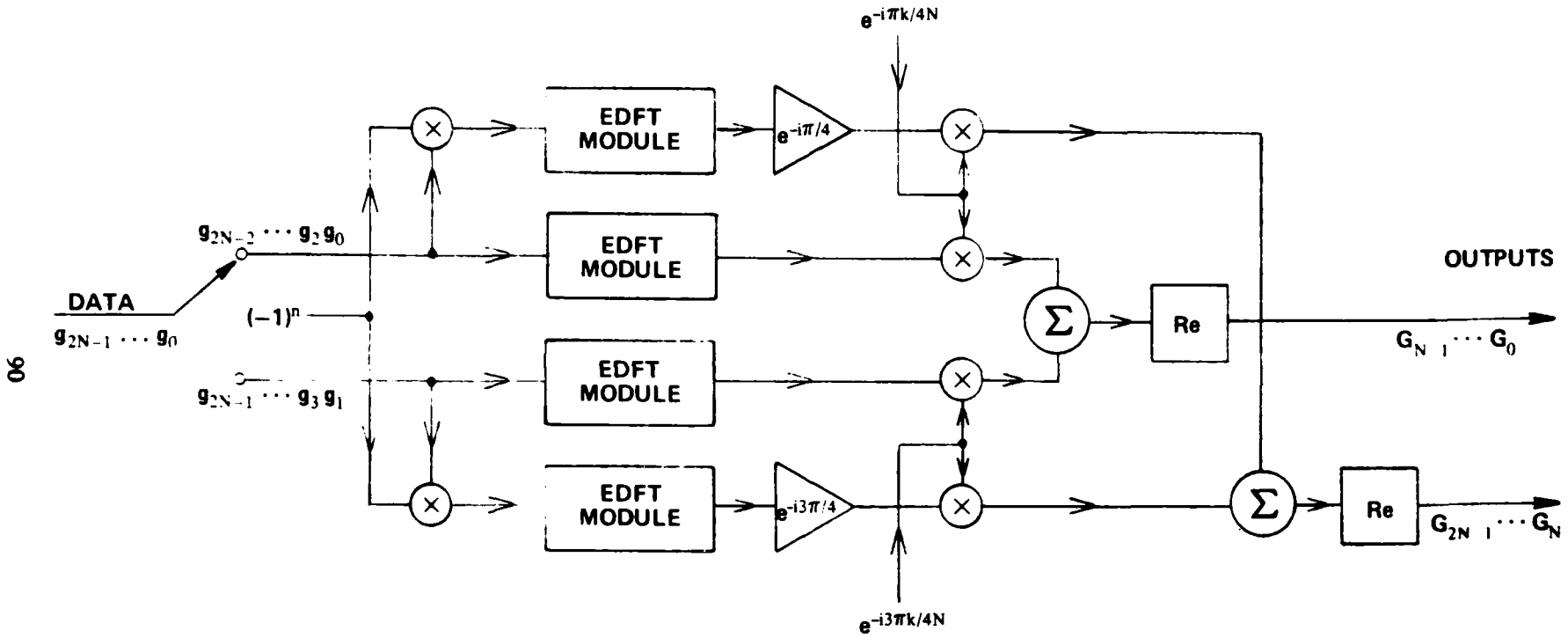


Fig. 2 EDCT of Size 2V Using Components (Filters and Chirp Generators) for an EDCT of Size N.

APPENDIX G

SIGNAL PROCESSING ARCHITECTURES USING TRANSVERSAL FILTER TECHNOLOGY

SIGNAL PROCESSING ARCHITECTURES USING TRANSVERSAL FILTER TECHNOLOGY*

H. J. Whitehouse, R. W. Means, and J. M. Speiser
Naval Undersea Center

INTRODUCTION

A large portion of the computational load for many signal processing problems consists of the computation of linear transforms. For time-invariant linear transforms such as cross convolution or matched filtering, the transversal filter provides a highly parallel computational module with high throughput and minimal control overhead [1]. This paper will show how similar computational modules can be configured to provide similar computational advantages for a large class of time-variant linear transforms including one-dimensional and multi-dimensional discrete Fourier transforms and one-dimensional and two-dimensional discrete cosine transforms. Furthermore, time-variant transform modules may be combined to implement high capacity time-invariant linear transforms. The implementation of these techniques using surface acoustic wave (SAW) and charge coupled device (CCD) technology permits the real-time solution of several important signal processing problems including image data compression, wideband radar signal analysis and spread spectrum communications.

COMPUTATIONAL MODULES

A linear transform on sampled data of finite extent may be viewed as the multiplication of a vector by a matrix. Multiplication by diagonal, circulant, or Toeplitz matrices may be accomplished rapidly with simple computational hardware modules. Multiplication by an N X N diagonal matrix requires only a scalar multiplier and a memory containing N values to provide serial access to the reference function Multiplication by an N X N Toeplitz matrix corresponds to a convolution and may be performed using a transversal filter having 2N−1 taps. Multiplication by an N X N circulant is a special case of multiplication by N X N Toeplitz matrix in which the length of the transversal filter may be reduced to N taps if the data block is recirculated through the filter or reread into the filter from a buffer memory.

ONE-DIMENSIONAL DFT

Linear filters have been used for many years for the calculation of the power spectra of continuous signals. One of the earliest methods used a bank of wave filters to measure the spectra in fractional octave bands for telephone network equalization [2]. However, when increased resolution was required the number of filters rapidly become unmanageable. An alternative which overcame the difficulty of a large number of filters each with small time-bandwidth product was to substitute one linear fm (chirp) filter with large time-bandwidth product and to employ matched filtering. In this system the signal to be analyzed is used to single sideband (SSB) modulate a locally generated chirp signal and the composite modulated signal is filtered in a chirp delay line matched filter. Each component of the input signal spectrum shifts the locally generated chirp to a different position in the spectrum after SSB modulation and these shifted chirps then correlate with the reference signal represented as the impulse response of the matched filter at different times. Thus the output signal amplitude-time history reflects the amplitude-frequency composition of the input signal.

*To be published in Proceedings of the 1975 IEEE International Symposium on Circuits and Systems (Boston, 21-23 April, 1975).

Bleustein [3] recognized that the discrete Fourier transform (DFT) of sampled data was amenable to a similar interpretation. In addition to just calculating the magnitude of the Fourier transform, linear filters could calculate the phase and thus all of the operations such a cross convolution and a cross correlation could be calculated. This technique came to be called the chirp-Z transform (CZT) and can be applied to other problems besides just the calculation of the DFT [4]. Prior to these developments, digital computation of the DFT had been significantly improved by the use of a special algorithm called the fast Fourier transform (FFT) which was described by Cooley and Tukey [5]. The FFT algorithm gained rapid popularity in signal processing since it allowed the calculation of the DFT to be done using significantly fewer machine operations (multiplications) than direct evaluation.

The DFT can be defined with many normalizations just as the continuous Fourier transform can be. For this paper the definition of the direct or forward transform of a complex vector g of length N is given as

$$G = Fg = \sum_{n=0}^{N-1} e^{-j\pi 2nm/N} g_n \tag{1}$$

and the inverse transform as

$$g = F^{-1} G = \frac{1}{N} \sum_{n=0}^{N-1} e^{j\pi 2nm/N} G_n \tag{2}$$

where F is an N × N matrix with elements $F_{n,m} = e^{-j2\pi nm/N}$.

By direct inspection it is observed that, if symmetries of the function $\exp j\pi 2nm/N$ are not exploited, then the number of complex multiplications required will be N^2 corresponding to N multiplications for each frequency component evaluated. Even on high speed digital computers this can become the limiting consideration in signal processing applications. The advantage of the FFT algorithm is that for highly composite values of the DFT size N the number of multiplications is proportional to $N \log_2 N$.

Although the FFT has been successful in substantially reducing the computing time and cost of using general purpose digital computers it has several disadvantages for special purpose real time computation. At high throughput rates which are required for real time image processing the processor either must operate $\log_2 N$ times faster than the data rate or pipeline structures which use distributed memory and $\log_2 N$ multipliers must be used. In addition, the internal arithmetic of the FFT processor must be done at increased precision in order to compensate for the multiple round off errors introduced by the successive stages in the FFT processor. Although these difficulties can be overcome, it is not always possible to arrange the computation in a form where the size of the transform is highly composite. For the above reasons and because of the difficulty of obtaining small, low power, fast analog to digital converters, linear transversal filter implementations of the chirp-Z-transform are attractive [6] rather than the previous CZT implementation which used an FFT to perform the required convolution.

The DFT may be easily reduced to the form suitable for linear filtering by the substitution

$$2nm = n^2 - (n-m)^2 + m^2 \tag{3}$$

which changes a product of variables into a difference so that

$$G_m = e^{-j\pi m^2/N} \sum_{n=0}^{N-1} e^{j\pi(n-m)^2/N} \; e^{-j\pi n^2/N} g_n \tag{4}$$

This form is seen to be equivalent to factoring the Fourier matrix F into the product of three matrices

$$F = DTD \tag{5}$$

where D is a diagonal matrix with elements $d_{nn} = \exp(-j\pi n^2/N)$ and T is a Toeplitz matrix with elements $T_{nm} = \exp(j\pi(n-m)^2/N)$.

The CZT algorithm is easily implemented by transversal filter techniques. In this case the DFT is computed by premultiplication by a discrete chirp, convolution with a discrete chirp, and postmultiplication by a discrete chirp. Figure 1 shows this configuration. However, it must be remembered that both the multiplications and convolutions are complex and a suitable representation of the complex numbers must be used. One representation is by real and imaginary part. Figure 2 shows the DFT organized as a CZT and implemented with parallel computation of the real and imaginary parts. In Figure 2 the input signal is represented as $g = g_R + jg_I$ and the output signal is represented as $G = G_R + jG_I$, where it is understood that $g = g_n$ $n = 0, \ldots, N-1$ and $G = G_n$ $n = 0, \ldots, N-1$.

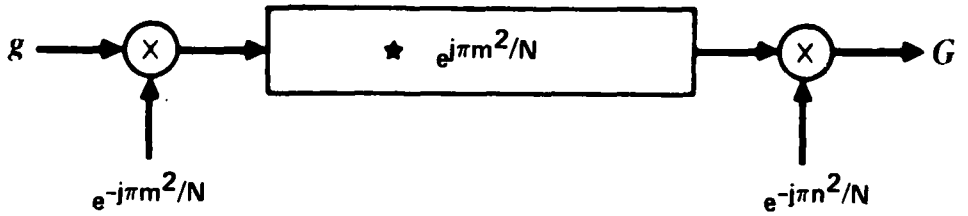
In order to determine the specific form of the transversal filters it is necessary to know the specific value of N. When N is odd the Toeplitz matrix T may be represented as a transversal filter with $2N-1$ complex taps $h_{-(N-1)}$ to h_{N-1} where $h_n = W^{-n^2/2}$, $n = -(N-1)$ to $(N-1)$, and $W = \exp(-j2\pi/N)$. The required convolution has been implemented with the general transversal filter shown in Figure 3.

When N is even, it can be shown that $T_{n,m} = T_{N+n,\,m}$ where the subscripts are reduced mod N. Thus T is a circulant matrix and can be implemented with a recirculating transversal filter as shown in Figure 4 where the number of complex taps is N and tap weights are: $h_n = W^{-n^2/2}$ $n = 0, \ldots, N-1$.

In the specific case when N is an odd prime, additional simplification is possible. It is possible to eliminate the multipliers. The DFT may be written as

$$G_o = \sum_{n=0}^{N-1} g_n \tag{6}$$

$$G_m - g_o = \sum_{n=1}^{N-1} e^{-j\pi 2nm/N} g_n \qquad \text{for } m = 1, \ldots, N-1 \tag{7}$$



★ Denotes either convolution or circular convolution

Figure 1. Chirp-Z-Transform Implementation of the DFT

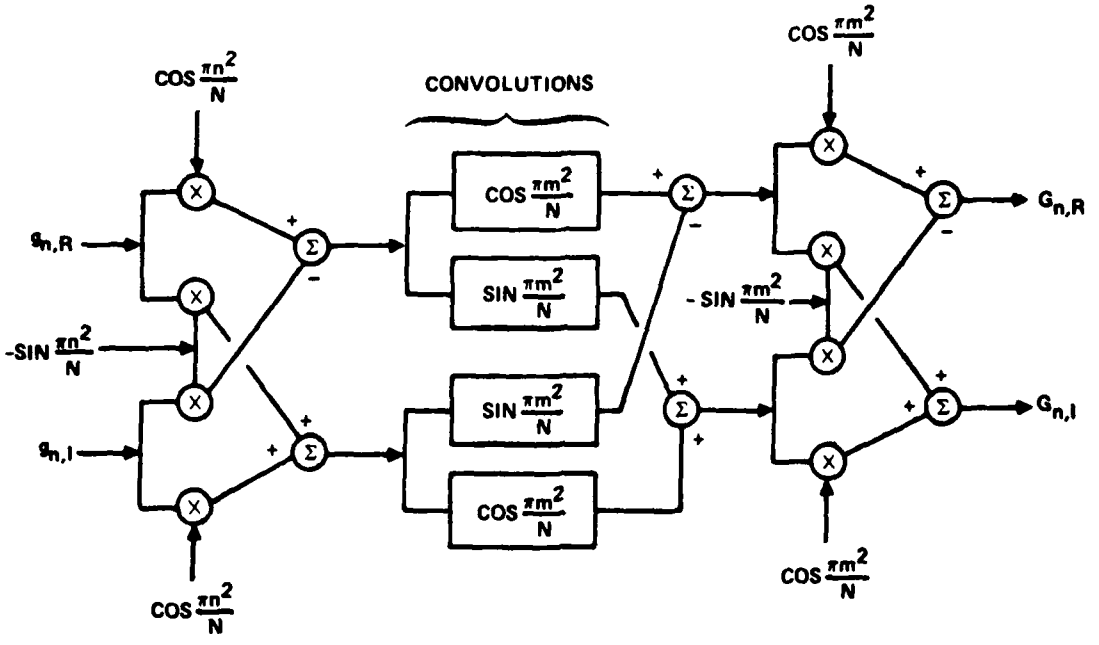


Figure 2. DFT Via CZT Algorithm with Parallel Implementation of Complex Arithmetic

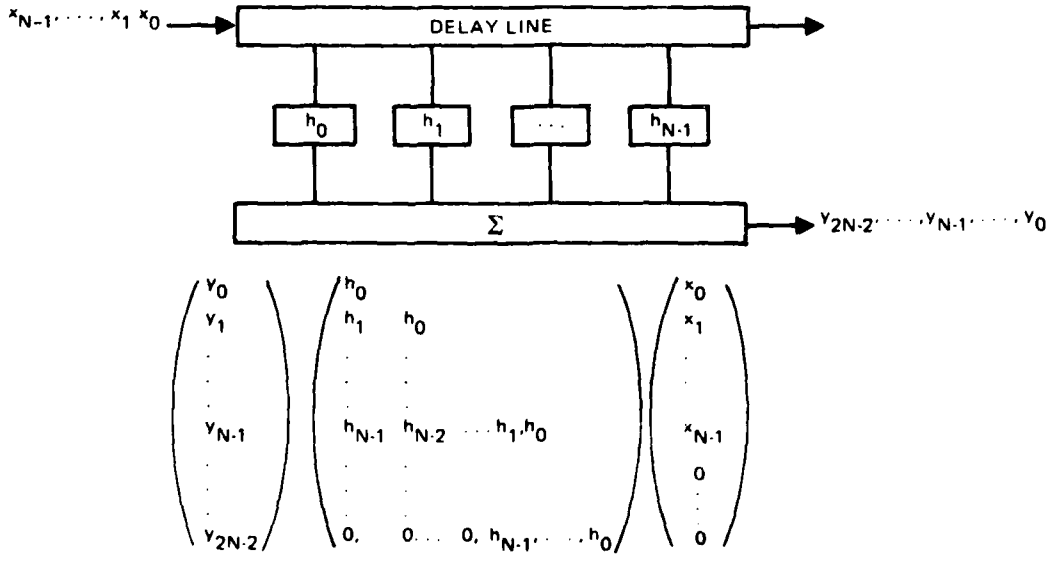


Figure 3. Transversal Filter

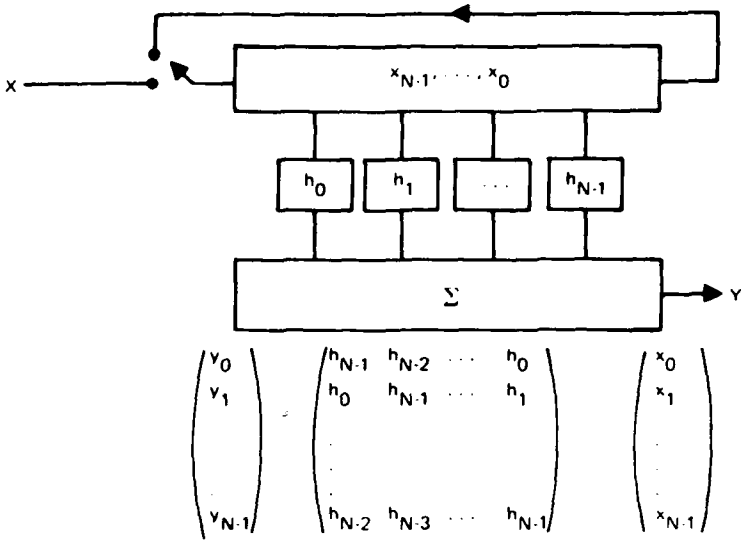


Figure 4. Circular Convolution

Since only nonzero values of n and m occur in the right hand side of Equation (7) it is possible to use Gauss's analogy between logarithms and indices with respect to a primitive root [7] to replace the product nm by a primitive root raised to a sum, thus reducing this computation to a correlation between permuted functions [8]. In matrix notation

$$G' - g_0 \, I = \mathbf{F}' g' \tag{8a}$$

where

$$I = \begin{bmatrix} 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{bmatrix} \tag{8b}$$

and the matrix $\mathbf{F}'$ can be factored into the three matrices

$$\mathbf{F}' \quad \mathbf{P}^t \mathbf{C} \mathbf{P}, \tag{9}$$

where $\mathbf{P}$ is a $N-1$ permutation matrix, $\mathbf{C}$ is a $(N-1 \times N-1)$ circulant matrix, $\mathbf{P}^t$ is the transpose of $\mathbf{P}$. G' and g' are column vectors of size $N-1$ derived from G and g by deleting, respectively, G_0 and g_0. The

elements of the matrix C are $C_{n,m} = F_{(n)_p(m)_p}$ n, m = 1, . . . , N − 1 and $(n)_p = r^n \bmod N$, where r is a primitive root of N and the elements of P are

$$P_{nm} = \delta_{(n)_p m} \qquad n, m = 1, \ldots N-1 \tag{10}$$

where δ_{km} is the Kronecher delta and p implies permuted.

Thus, for the case where N is an odd prime, C is a circulant matrix and can be implemented with a recirculating transversal filter whose tap weights are

$$h_n = C_{n,1} \qquad n = 1 \ldots N-1 \tag{11a}$$

$$= W^{(n)_p} \qquad n = 1, \ldots N-1 \tag{11b}$$

and the Fourier transform coefficients are given by

$$G_o = \sum_{n=0}^{N-1} g_n \tag{12a}$$

and

$$G_m = G'_m + g_o. \qquad m = 1 \ldots N-1 \tag{12b}$$

These concepts are illustrated in Figure 5 for N = 5. Thus, the one-dimensional architectures may be summarized as a time-varying operation, a convolution, and a second time-varying operation where for arbitrary N the time-varying operations are multiplication by a diagonal matrix; and where for the special case, N is an odd prime, the time-varying operations are multiplication by a permutation matrix and its transpose.

$$F = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & W^1 & W^2 & W^3 & W^4 \\ 1 & W^2 & W^4 & W^1 & W^3 \\ 1 & W^3 & W^1 & W^4 & W^2 \\ 1 & W^4 & W^3 & W^2 & W^1 \end{pmatrix} \qquad F' = \begin{pmatrix} W^1 & W^2 & W^3 & W^4 \\ W^2 & W^4 & W^1 & W^3 \\ W^3 & W^1 & W^4 & W^2 \\ W^4 & W^3 & W^2 & W^1 \end{pmatrix}$$

$2^1 \equiv 2 \bmod 5 \qquad 2^2 \equiv 4 \bmod 5 \qquad 2^3 \equiv 3 \bmod 5 \qquad 2^4 \equiv 1 \bmod 5$

$$P^t = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \qquad C = \begin{pmatrix} W^4 & W^3 & W^1 & W^2 \\ W^3 & W^1 & W^2 & W^4 \\ W^1 & W^2 & W^4 & W^3 \\ W^2 & W^4 & W^3 & W^1 \end{pmatrix} \qquad P = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Figure 5. Example of the Prime Transform when N = 5

IMPLEMENTATION

Many types of transversal filter implementations may be easily accomplished using analog sampled data techniques [9]. Each offers some advantages and some disadvantages in any particular application, although they are all architecturally similar. In this paper the implementations to be discussed are surface acoustic wave (SAW) devices, charge coupled devices (CCD) and hybrid analog-digital correlators.

SURFACE ACOUSTIC WAVE DEVICES

Surface Acoustic Wave (SAW) devices can accept either analog or sampled analog input and the output is analog. A substrate of piezoelectric material is polished on one face and a pattern of aluminum or other conductor is deposited by photolithographic techniques. In its simplest configuration, sets of interdigitated finger electrodes are spaced at the sampling rate distance through the use of the relationship $d_s = C_R t_s$, where d_s is the tap spacing, C_R is the Rayleigh wave velocity, and t_s is the sampling increment. For a typical substrate of ST-cut quartz, $C_R \sim 3$mm/μsec and for a typical sampling increment of 150 nsec, $d_s = 0.45$ mm and a 150 point DFT can be implemented by recirculating convolution in an active length less than 75 mm (3 in.) on a 100 mm (4 in.) substrate. A simple real transversal filter is shown in Figure 6 along with the method of establishing the tap weights. A prototype complex filter [10] is shown packaged in Figure 7. A similar device has been used as the premultiplication and postmultiplication reference function generator and double-balanced mixers have been employed as the multipliers.

Physical limitations on the size of available substrates limit the available DFT size. For an active length of substrate $L_s = 150$ mm and a surface wave velocity $C_R = 3$ mm/μsec the maximum transform size N_{max} for a data rate F_s is approximately $N_{max} = L_s F_s / C_R \approx 50\, F_s$ with F_s in megaHertz. Current SAW

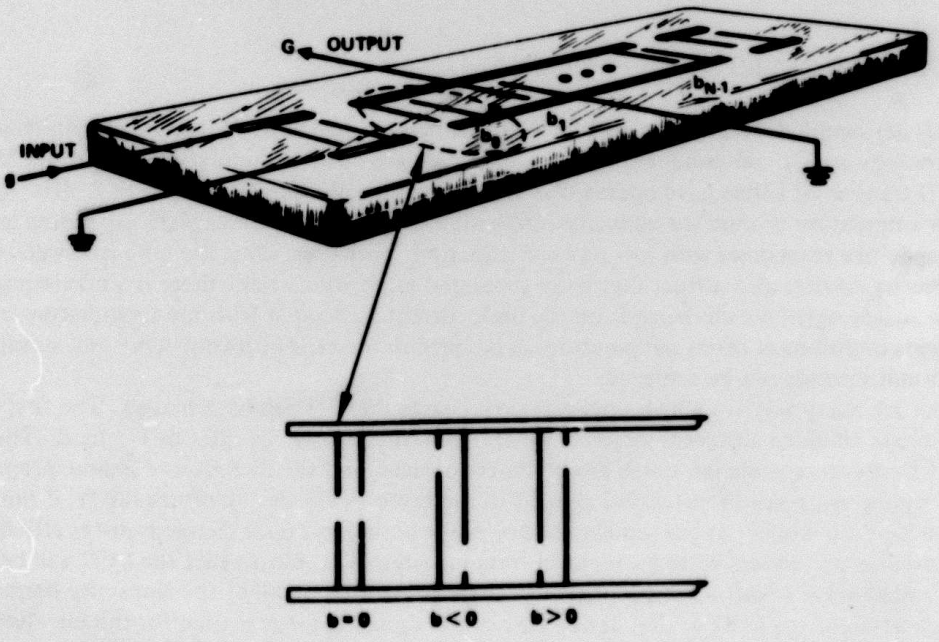
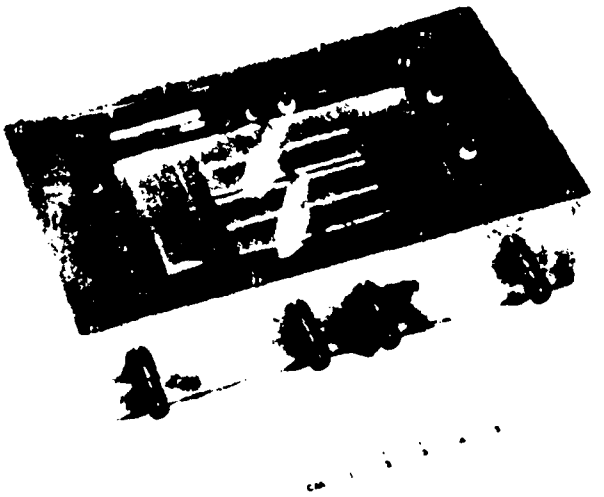


Figure 6. Surface Wave Transversal Filter



CZT TRANSVERSAL FILTER

Figure 7. 32 Complex Tap SAW CZT Filter

transversal filters operate at sample rates from 1 to 100 MHz with carrier frequencies from 5 to 500 MHz with typical fractional bandwidths of about 10%. Although there are some piezoelectric semiconductors, silicon is nonpiezoelectric. Heteroepitaxial materials such as Aluminum Nitride (AlN) and Silicon (Si) cogrown on a sapphire substrate are currently under development for monolithic integrated circuits where The AlN is the piezoelectric and Si is the semiconductor. Such monolithic circuits should make possible monolithic CZTs at 100 to 500 MHz data rates.

CHARGE COUPLED DEVICES

CCDs are sampled data analog circuits which can be fabricated by Metal Oxide Semiconductor (MOS) technology as LSI components [11]. As such they are directly compatible with other MOS circuits. Current CCD transversal filters have operated as video devices with sample rates up to 5 MHz. CCDs operate by the manipulation of injected minority carriers in potential wells under MOS capacitors and thus behave as capacitive reactances with low power dissipation. However, since the potential wells which contain the minority carriers also attract thermally generated minority carriers, there is a maximum storage time for the analog signal which depends on the dark current associated with the temperature of the silicon. Under normal conditions at room temperature, dark currents are tens of nAmps/cm^2 and storage times of hundreds of milliseconds can be achieved.

There are many ways in which unidirectional charge transfer can be achieved. The first developed was a three-phase clocking structure which is illustrated in the transversal filter of Figure 8. The three electrode CCD structure is planar, much like the SAW devices, and the direction of charge propagation is determined by the sequence of potentials applied to the three electrodes. Unfortunately, if the minority carriers are allowed to collect at the semiconductor-oxide boundary, poor charge transfer efficiency will result due to minority carriers getting caught in trapping sites. This means that the CCD will behave nonlinearly unless there is sufficient propagating charge present to fill all of the traps. By biasing the operating condition of the CCD so that about 10% of the dynamic range is used for the injection of a "fat zero," the traps are kept continuously filled and the device has over a 60 dB dynamic range. In practice,

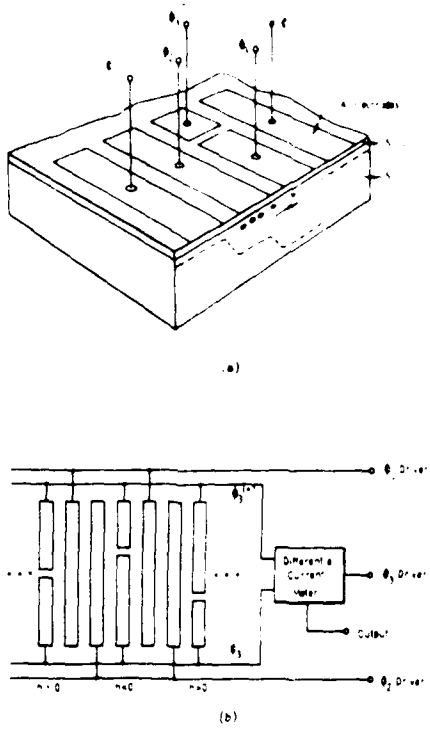


Figure 8 Schematic of the Sampling, Weighting, and Summing Operation

a video signal representing the signal to be processed is added to a fixed bias somewhat larger than one-half of the peak-to-peak value of the signal. Since the effective storage time of the device is long relative to the time required to execute a convolution, CCDs can be considered to be interruptible signal processors and as such are more compatible with the executive control required for signal processing. A 64 point CCD filter with discrete cosine transform sine and cosine chirps is shown in Figure 9. This chip was developed by Texas Instruments for the Naval Undersea Center for image processing. The discrete cosine transform is described in a subsequent section.

Current research in CCDs is directed toward improving the charge transfer efficiency and removing the requirement of continuous "fat zero" charge injection by ion implantation techniques which keep the minority carriers away from the semiconductor oxide boundary. Ion implantation is also being used to provide asymmetric potential wells so that simpler two-phase clocking can be employed. Currently available CCDs have 500 stages with 0.9999 transfer efficiency and devices with up to 2000 stages are planned.

Another charge transfer device similar to the CCD is the Bucket Brigade Device (BBD). This is a sequence of MOS transistors coupled together by diffusion enhanced Miller capacitance. Although these devices do not operate at frequencies as high as CCDs, they have better low frequency performance since they include active devices. A CZT has been implemented with two BBD chips. Two 200 tap filters are implemented on each chip: one a discrete cosine and the other a discrete sine filter. The device, the complex chirp used in the premultiplier and a typical input and output are shown in Figure 10. The input is an offset cosine wave and the output shows a D.C. component plus a response at the cosine wave frequency. These filters can operate at 100 kHz and have tap accuracies better than 1%. With careful control of geometry, both BBD and CCD filters with tap accuracies approaching 0.1% should be possible. This chip was also developed by Texas Instruments for the Naval Undersea Center.

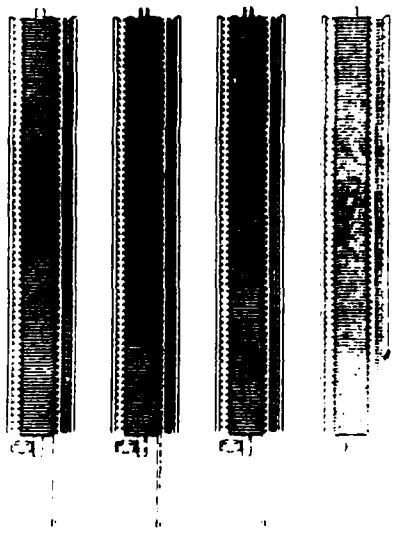


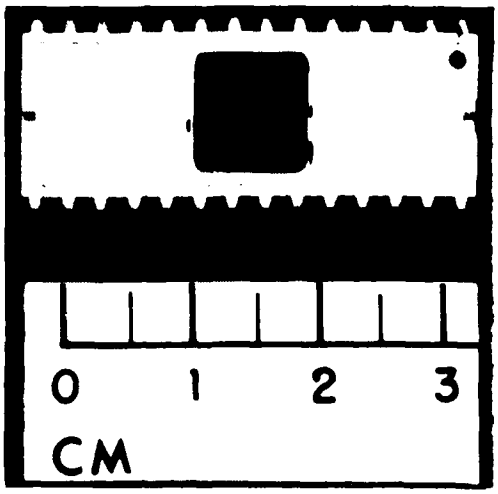
Figure 9. 64 Point CCD Filters

A hybrid binary-analog CCD correlator module with 32 analog taps [12] has been made for the Naval Undersea Center by General Electric and is shown diagrammatically in Figure 11. The maximum clock rate of the module is 4 MHz. These modules may be cascaded to increase the correlation length to provide an analog versus multilevel cross correlator with analog output. The correlator module uses charge propagation through only 3 stages since the CCD is arranged in 32 stages of 3 samples each. By transverse shifting of the charge instead of longitudinal shifting, charge transfer inefficiency degradation is avoided and the modules can be cascaded to very large sizes. However, with this configuration, dark current non-uniformity must be controlled to prevent time variable pattern noise from contributing to the output. Systematic errors in the tap weights can be measured and stored and the analog signal corrected for the measured variation in uniformity before it is stored in the CCD registers.

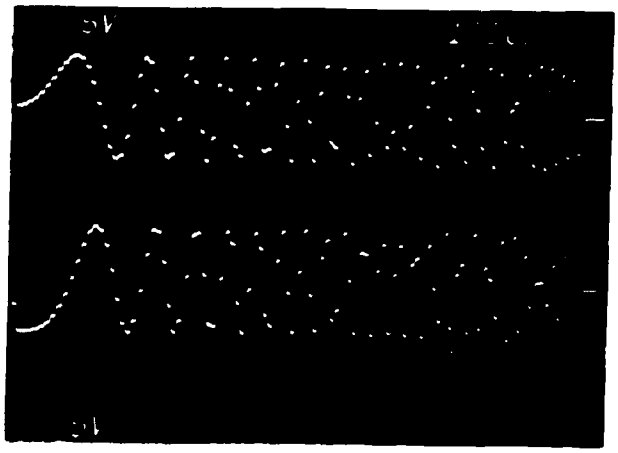
DISCRETE COSINE TRANSFORM

Closely related to the DFT is the discrete cosine transform (DCT). Two different types of DCTs are useful for reduced redundancy television image transmission. Both are obtained by extending a length N real data block to have even symmetry, taking the discrete Fourier transform (DFT) of the extended data block, and saving N terms of the resulting DFT. Since the DFT of a real, even sequence is a real, even sequence, either DCT is its own inverse if a normalized DFT is used.

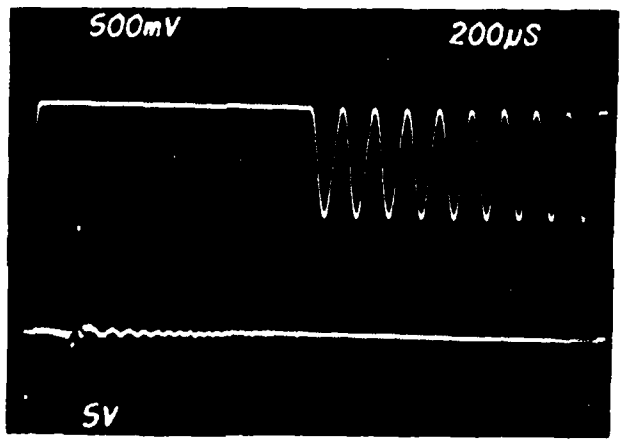
The "Odd DCT" (ODCT) extends the length N data block to length 2N-1, with the middle point of the extended block as a center of even symmetry. The "Even DCT" (EDCT) extends the length N data block to length 2N, with a center of even symmetry located between the two points nearest the middle.



BBD F.



PREMULTIPLIER INPUT



INPUT TO DCT

DCT OUTPUT

Figure 10. BBD Performance

For example, the odd length extension of the sequence $g_0g_1g_2$ is $g_2g_1g_0g_1g_2$, and the even length extension is $g_2g_1g_0g_0g_1g_2$. In both cases, the symmetrization eliminates the jumps in the periodic extension of the data block which would occur if one edge of the data block had a high value and the other edge had a low value: in effect it performs a sort of smoothing operation with no loss of information. It will be noted that the terms "odd" and "even" in ODCT and EDCT refer only to the length of the extended data block — in both cases the extended data block has even symmetry. Both types of DCT may be implemented using compact, high speed, serial access hardware, in structures similar to those previously described for the chirp-Z transform (CZT) implementation of the DFT.

Let the data sequence be $g_0, g_1, \ldots, g_{N-1}$. The ODCT of g is defined as

$$G_k = \sum_{n=-(N-1)}^{N-1} g_n \, e^{\frac{-j2\pi nk}{2N-1}} \quad \text{for } k = 0, 1, \ldots, N-1 \tag{13}$$

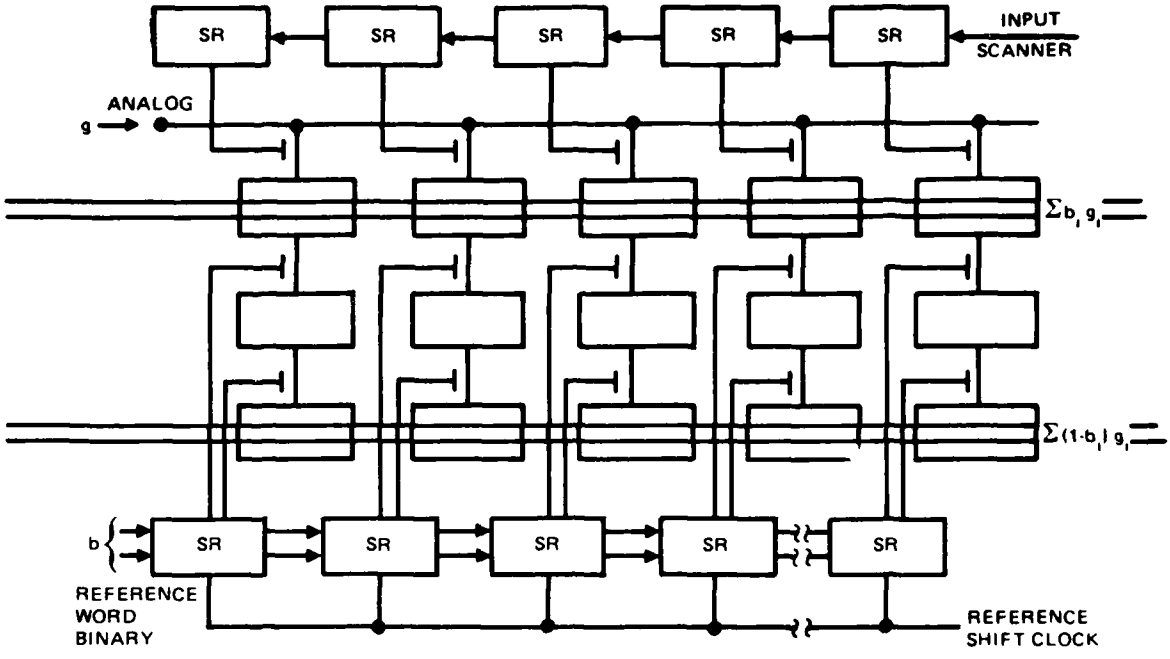


Figure 11. Schematic Diagram of a Binary-Analog CCD Correlator Module

where $g_{-n} = g_n$ for n = 0, 1, . . . , N - 1. By straightforward substitution it may be shown that

$$G_k = 2\,\mathrm{Re} \sum_{n=0}^{N-1} \tilde{g}_n \; e^{\frac{-j2\pi nk}{2N-1}} \tag{14a}$$

where $\tilde{g}$ is defined by equation (14a)

$$\tilde{g}_n = \begin{cases} 0.5\, g_0, & n = 0 \\ g_n, & n = 1, \ldots, N-1 \end{cases} \tag{14b}$$

The identity (15) may be used to obtain the CZT form of the ODCT shown in equation (16).

$$2nk = n^2 + k^2 - (n-k)^2 \tag{15}$$

$$G_k = 2\,R_e \left\{ e^{\frac{-j\pi k^2}{2N-1}} \sum_{n=0}^{N-1} e^{\frac{-j\pi n^2}{2N-1}} \tilde{g}_n \; e^{\frac{j\pi (n-k)^2}{2N-1}} \right\} \tag{16}$$

The EDCT of g is defined by equation (17a), where the extended sequence is defined by equation (17b).

$$G_k = e^{\frac{-j\pi k}{2N}} \sum_{n=-N}^{N-1} g_n \; e^{\frac{-j2\pi nk}{2N}} \quad \text{for } k = 0, 1, \ldots, N-1 \tag{17a}$$

$$g_{-1-n} = g_n \quad \text{for } n = 0, 1, \ldots, N-1 \tag{17b}$$

If the mutually complex conjugate terms in equation (17a) are combined, then equation (18) results. Equation (18) may be viewed as an alternate way of defining the EDCT.

$$G_k = 2\,\mathrm{Re}\left\{ e^{\frac{-j\pi k}{2N}} \sum_{n=0}^{N-1} g_n \; e^{\frac{-j2\pi nk}{2N}} \right\} \tag{18a}$$

$$G_k = 2 \sum_{n=0}^{N-1} g_n \cos\left[\frac{2\pi(n+0.5)k}{2N}\right] \tag{18b}$$

Equation (18) may be put in the CZT format as

$$G_k = 2\,\mathrm{Re}\left\{ e^{\frac{-j\pi k}{2N}} \; e^{\frac{-j\pi k^2}{2N}} \sum_{n=0}^{N-1} g_n \; e^{\frac{-j\pi n^2}{2N}} \; e^{\frac{j\pi(n-k)^2}{2N}} \right\} \tag{19}$$

Ahmed [13] has investigated the use of the EDCT as a substitute for the Karhunen-Loeve transform for exponentially correlated data and finds that it is superior to the Fourier transform and is comparable to the Karhunen-Loeve (K-L) in rate-distortion performance while maintaining the computational simplicity of a transform which does not depend on the picture statistics. Habibi [14] has shown by simulation that the DCT is equivalent in a mean-square-error sense to the K-L transform under basis restriction. A 100 point ODCT for use in a 10 frame per second experimental TV image compression system has been constructed by the Naval Undersea Center for the Advanced Research Projects Agency (ARPA) using bucket brigade device transversal filters. The transversal filter has 199 nonzero taps and both the cosine filter and sine filter required for complex arithmetic are implemented on a single chip. A microphotograph of the filters is shown in Figure 12.

TWO-DIMENSIONAL DFT

The DFT of a two-dimensional array, $g_{n_1 n_2}$ may be computed by successive applications of the one-dimensional DFT. This concatenation is readily observed by writing the expression for the two-dimensional DFT as

$$G(k_1, k_2) = \sum_{n_2=0}^{N_2-1} \left\{ \sum_{n_1=0}^{N_1-1} g(n_1, n_2) \; e^{\frac{-j2\pi n_1 k_1}{N_1}} \right\} e^{\frac{-j2\pi n_2 k_2}{N_2}} \tag{20}$$

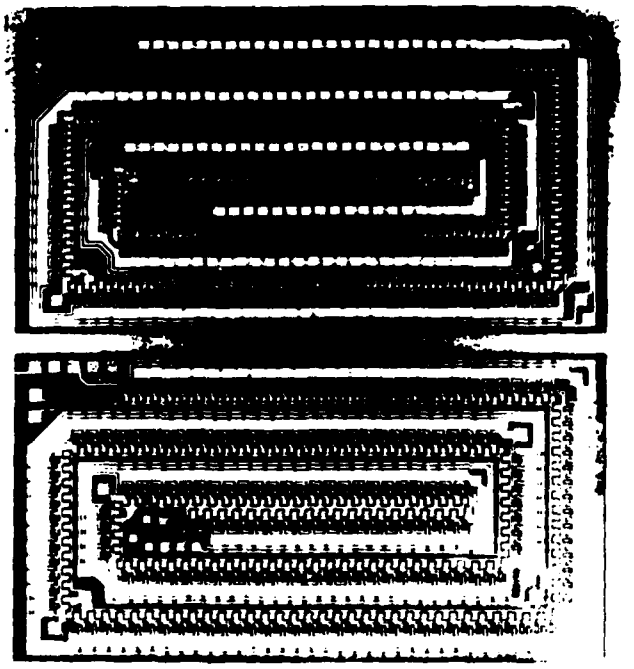


Figure 12. Bucket Brigade Chirp Filters

In two dimensions the concatenation property of the DFT can be exploited along with auxiliary memory to compute the two-dimensional transform by successively computing the CZT of rows of the input signal matrix and then using the auxiliary memory as a row to column transformation, i.e., transposing the partial Fourier transform matrix, and computing the two-dimensional Fourier transform with a second CZT.

An alternative method of computing the two-dimensional DFT is to use linear congruential scanning of the data. The two-dimensional discrete Fourier transform system will use an input scanning device and a one-dimensional discrete Fourier transform device as shown in Figure 13. The two-dimensional transform block size N_1 by N_2 is chosen such that N_1 and N_2 are relatively prime integers (i.e., they must have no common divisor). The one-dimensional Fourier transform device has a block length of $N = N_1 N_2$. The purpose of the input scanning device is to so order the input data that the one-dimensional Fourier transform of the length $N_1 N_2$ serial data string is identical to an N_1 by N_2 two-dimensional Fourier transform of the N_1 by N_2 input data samples. If desired, an output scanning device may also be used to provide the transform output points in normal order. The required scan may be derived from the representation [15] of a one-dimensional discrete Fourier transform matrix as a direct product matrix. The one-dimensional DFT in equation (21) is equivalent to the two-dimensional DFT in equation (22) when $N = N_1 N_2$ and N_1, N_2 are relatively prime

$$F_k = \sum_{n=0}^{N-1} f_n \, e^{\frac{-j2\pi nk}{N}} \quad \text{for } n, k = 0, 1, \ldots, N-1 \tag{21}$$

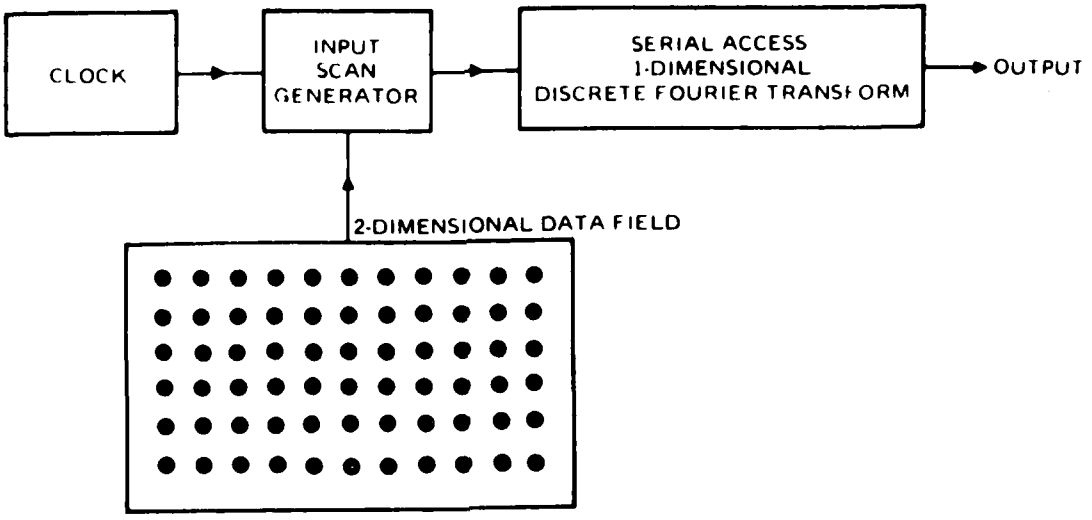


Figure 13. Two-dimensional Discrete Fourier Transform Device

$$G(k_1, k_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} g(n_1, n_2)\, e^{-j2\pi \left[\frac{n_1 k_1}{N_1} + \frac{n_2 k_2}{N_2}\right]} \tag{22}$$

for $n_1, k_1 = 0, 1, \ldots N_1 - 1$ and $n_2 k_2 = 0, 1, \ldots N_2 - 1$

In order to make the two transforms equivalent, it is necessary to find a pair of one-to-one functions $n(n_1, n_2)$ and $k(k_1, k_2)$ such that

$$\frac{nk}{N} = \frac{n_1 k_1}{N_1} + \frac{n_2 k_2}{N_2} \quad \text{(Modulo 1)} \tag{23}$$

or

$$nk = n_1 k_1 N_2 + n_2 k_2 N_1 \quad \text{(Modulo } N_1 N_2\text{)} \tag{24}$$

This may be accomplished by letting

$$n(n_1, n_2) = n_1 N_2 + n_2 N_1 \quad \text{(Modulo N)} \tag{25}$$

$$k(k_1, k_2) = k_1 U_1 N_2 + k_2 U_2 N_1 \quad \text{(Modulo N)} \tag{26}$$

where the constants U_1 and U_2 are the solutions of

$$N_2 U_1 = 1 \quad \text{(Modulo } N_1\text{)} \tag{27}$$

$$N_1 U_2 = 1 \quad \text{(Modulo } N_2\text{)} \tag{28}$$

Equations (27) and (28) will have solutions if and only if N_1 and N_2 are mutually prime. Under this condition, the mappings described by equations (25) and (26) will satisfy the requirement of equation (23). The linear congruential scan prescribed by Equation (25) may also be used to perform two-dimensional convolution or crosscorrelation using an ordinary one-dimensional transversal filter or crosscorrelator.

TWO-DIMENSIONAL CZT

Once the data is in a two-dimensional format, with simultaneous serial access to all the rows, it may be transformed in the "horizontal" direction by the structure shown in Figure 14. The individual one-dimensional discrete chirp filters and discrete chirp generator of Figure 14 may be implemented using CCDs or hybrid digital correlators. An acoustic surface wave device with multiple input taps may be used to access a column of the partially transformed output in a single shift time of the partial transform device. With appropriate coding of the surface wave column access device, it may also perform the discrete chirp premultiplication and the discrete chirp convolution of a DFT in the "vertical" direction. A complete two-dimensional CZT architecture is shown in Figure 15, and the required coding for the column access wave device is shown in Figure 16. The complex arithmetic may be implemented as described previously. A balanced mixer may be used for the fast multiplier required for the vertical transform. Lower speed variable transconductance multipliers may be used in the horizontal partial transform.

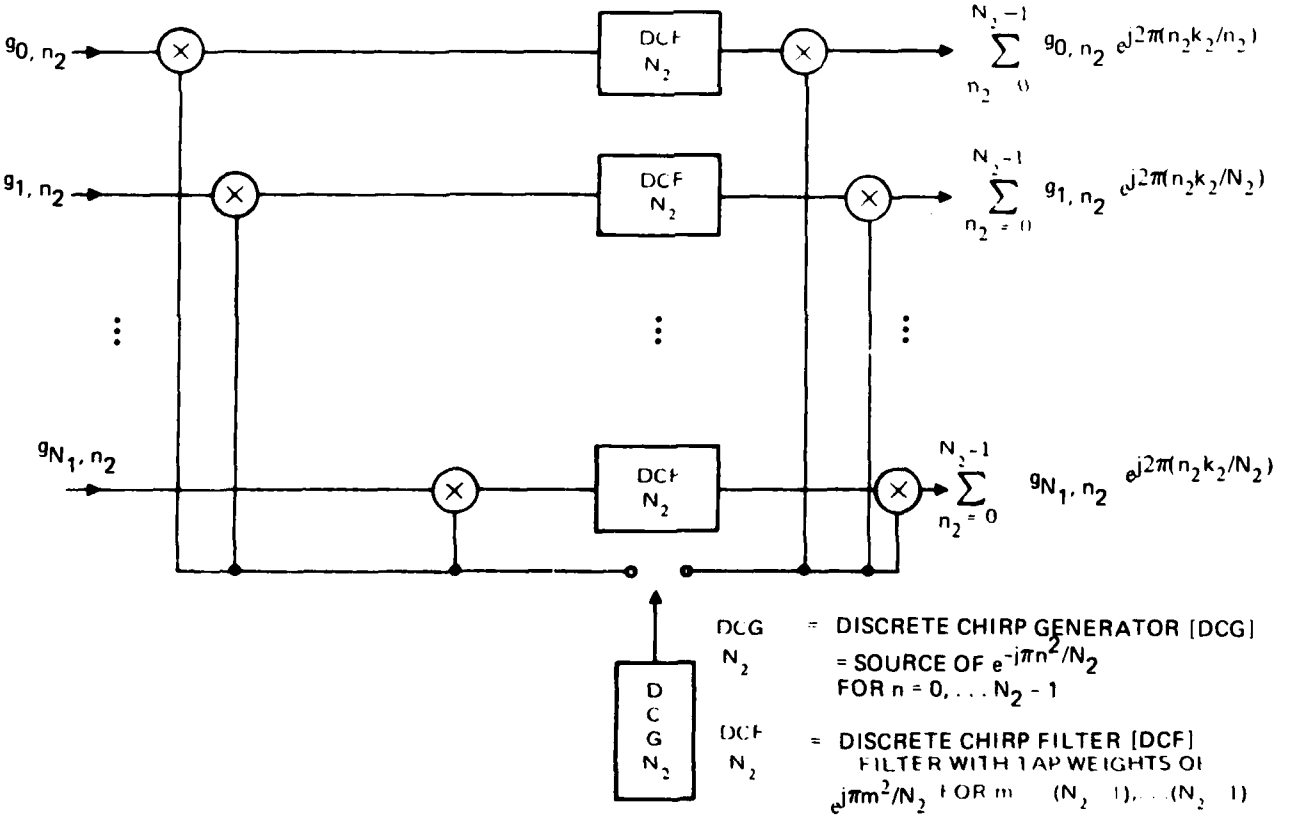


Figure 14. Two-dimensional Partial Chirp-Z Transform

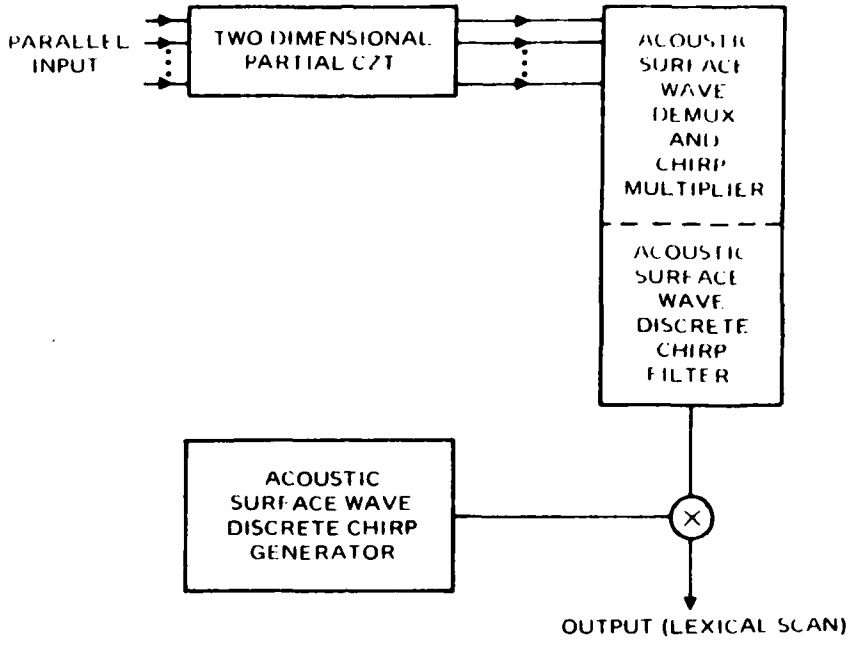


Figure 15. Hybrid Implementation of Two-dimensional CZT.

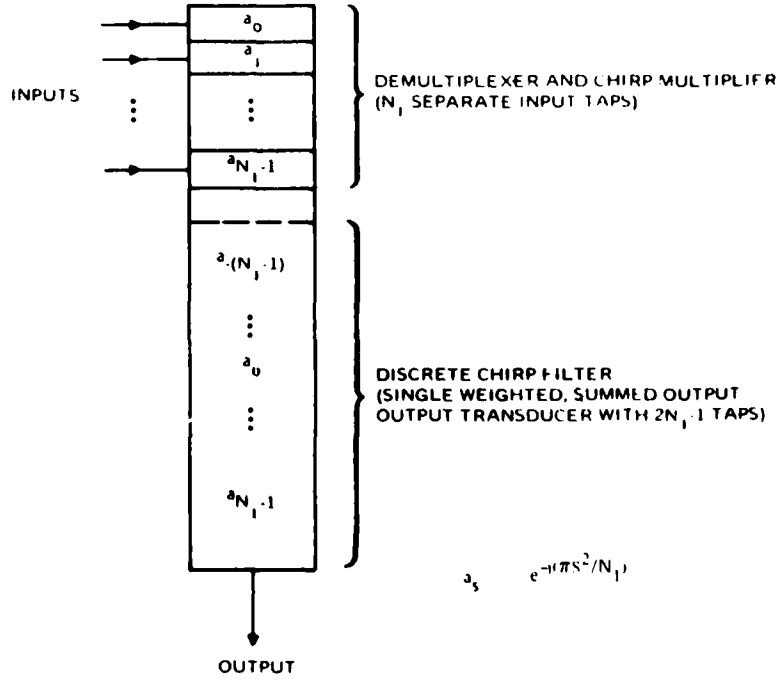


Figure 16. Tap Weights and Structure for Acoustic Surface Wave Combined Demultiplexer, Chirp Multiplier, and Discrete Chirp Filter

MODULAR CZT

Two methods for combining N_2 chirp-Z transform (CZT) modules of length N_1 to perform a discrete Fourier transform (DFT) of length N_1N_2 are described. The first method uses an auxiliary parallel-input, parallel-output DFT device of size N_2 and allows the transform of size N_1N_2 to be performed in the same time required for a single CZT module to perform a size N_1 transform. The second method uses an auxiliary parallel-input, serial-output DFT device of size N_2. If the second method is implemented entirely in a single technology, such as with CCDs, it performs the size N_1N_2 transform in N_2 times the amount of time required for a single CZT module to perform a size N_1 transform; if N_2 is a composite number, say $N_2 = M_1M_2$, the second method also permits the same hardware to perform M_1 simultaneous transforms of length N_1M_2.

A one-dimensional discrete Fourier transform may be written as a partial transform of a doubly subscripted representation of the data, followed by a pointwise multiplication, followed by a second partial transform as shown in equations (29) - (34):

$$G_k = \sum_{n=0}^{N-1} g_n \, e^{\frac{-j2\pi kn}{N}} \quad \text{for } k = 0, \ldots, N-1 \text{ and } N = N_1M_2 \tag{29}$$

Let

$$n = n_1N_2 + n_2 \quad \text{for } k_1, n_1 = 0, \ldots, N_1 - 1 \tag{30}$$

$$k = k_1 + k_2N_1 \quad \text{for } k_2, n_2 = 0, \ldots, N_2 - 1 \tag{31}$$

$$G_{k_1+k_2N_1} = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} g_{n_1N_2+n_2} \, e^{\frac{-j2\pi (k_1 + k_2N_1)(n_1N_2 + n_2)}{N_1N_2}} \tag{32}$$

$$G_{k_1+k_2N_1} = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} g_{n_1N_2+n_2} \, e^{\frac{-j2\pi k_1n_1}{N_1}} \, e^{\frac{-j2\pi k_2n_2}{N_2}} \, e^{\frac{-j2\pi k_1N_2}{N_1N_2}} \tag{33}$$

$$G_{k_1+k_2N_1} = \sum_{n_2=0}^{N_2-1} e^{\frac{-j2\pi k_2n_2}{N_2}} \, e^{\frac{-j2\pi k_1n_2}{N_1N_2}} \sum_{n_1=0}^{N_1-1} g_{n_1N_2+n_2} \, e^{\frac{-j2\pi k_1n_1}{N_1}} \tag{34}$$

Figures 17 and 17a show modular CZT implementations which follow from equation (34). The individual CZT subsystems shown in Figures 17 and 17A would be similar to the CZT implementations previously described. The parallel DFT required for the second partial transform in Figure 17 may be implemented as combination of summers and attenuators. This is shown in Figure 18 for $N_2 = 2$. In general, the attenuation factors are complex. A complete double length CZT is shown in Figure 19. Unfortunately, a parallel DFT implementation of this type becomes unwieldly if the dimension N_2 is very large.

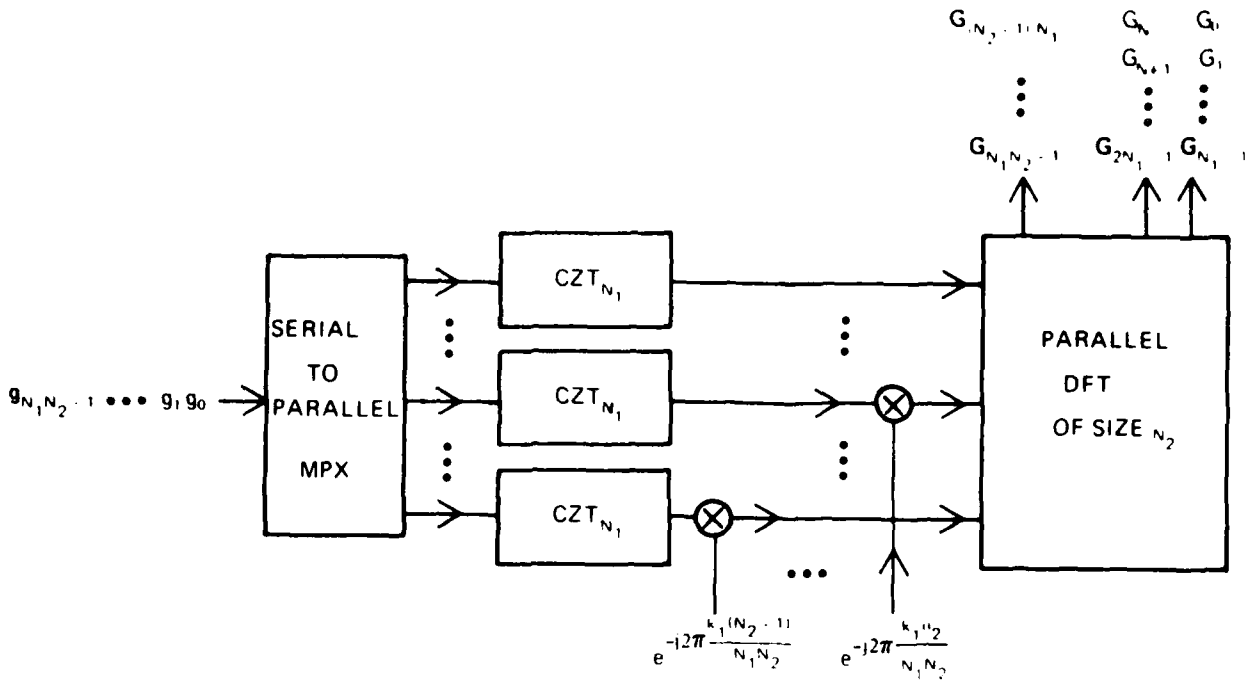


Figure 17. Organization of Modular CZT

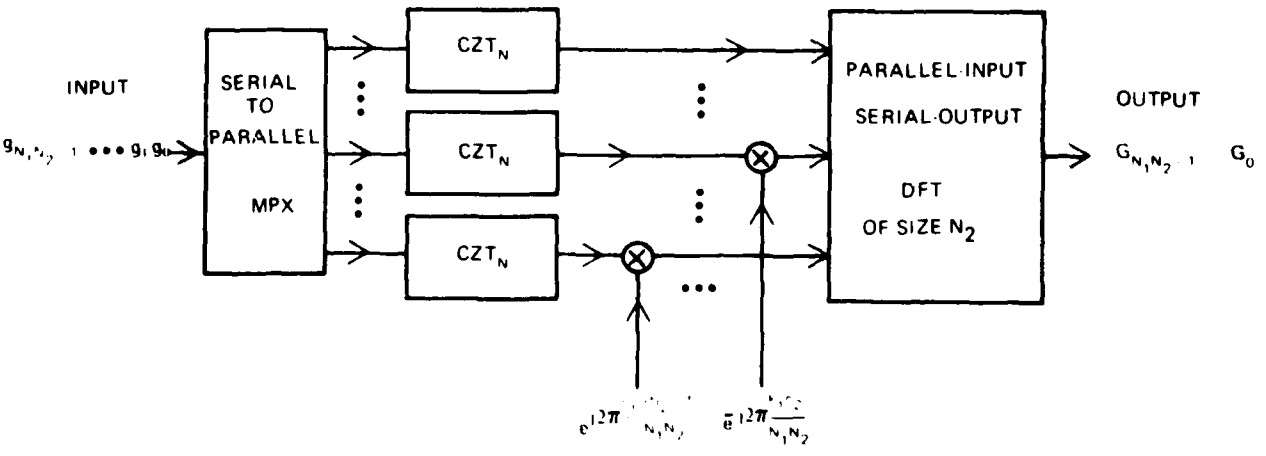


Figure 17A Alternate Organization of Modular CZT

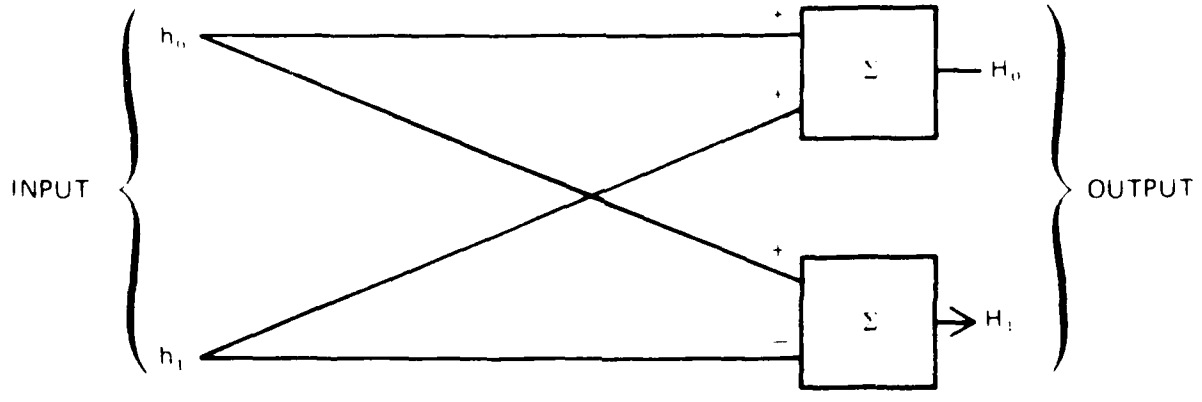


Figure 18. Parallel DFT of Size M = 2

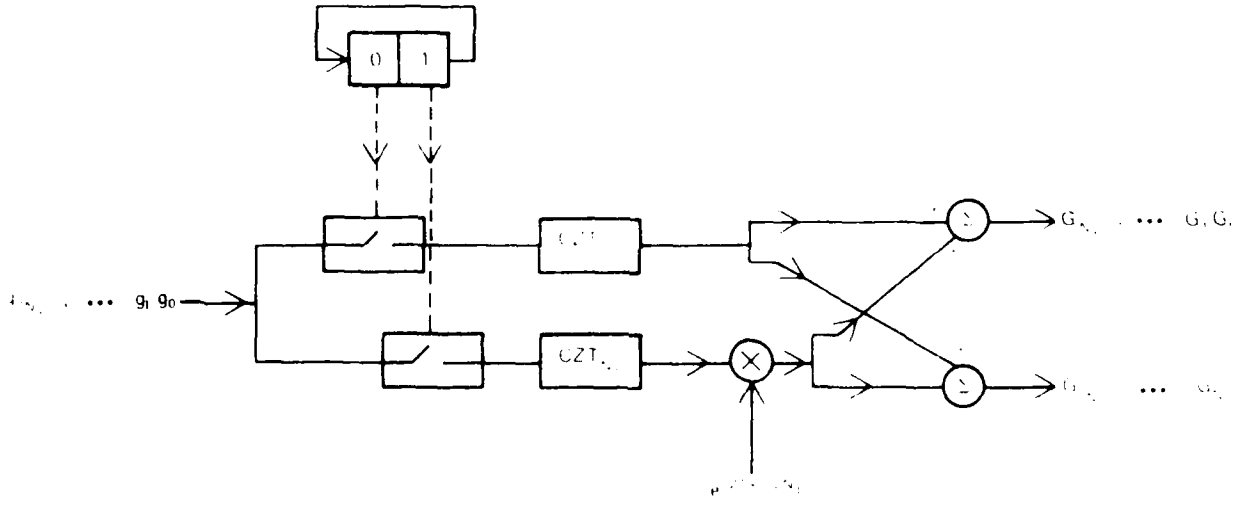


Figure 19. Double Length CZT

Other types of discrete Fourier transform implementations may be derived from the identities of equations (35) - (37)

$$H_k = \sum_{n=0}^{M-1} e^{\frac{-j2\pi kn}{M}} h_n \tag{35}$$

$$kn = \frac{1}{4}\left\{(k+n)^2 - (k-n)^2\right\} \tag{36}$$

$$H_k = \sum_{n=0}^{M-1} e^{\frac{-j\pi(k+n)^2}{2M}} e^{\frac{j\pi(k-n)^2}{2M}} h_n \quad \text{for } k = 0, \ldots, M-1 \tag{37}$$

These equations have been used to design a transform device in which signals are shifted through two CCD delay lines at different speeds. Alternatively, if the factors in equation (37) are interpreted as two waves propagating in opposite directions relative to the function to be transformed, it may be seen that the structure of Figure 20 also performs a discrete Fourier transform with speed comparable to that of a CZT. A surface wave device module which implements the triple product convolutional equation (37) has been built by Reeder [16] at United Aircraft. A schematic is shown in Figure 21.

TWO-DIMENSIONAL DCT

A two-dimensional DCT may be computed as a two-dimensional Fourier transform of a symmetrically extended data block. In order to minimize the required filter length, the scanning of the data block which has been symmetrized for the two-dimensional ODCT will now be considered in detail. The data block is extended using the double mirror symmetry defined in equation (38).

$$g(\pm n_1, \pm n_2) = g(n_1, n_2) \tag{38}$$

The two-dimensional DFT of the extended data block defines the two-dimensional DCT of the original data block given in equation (39), with $M_1 = 2N_1 - 1$, $M_2 = 2N_2 - 1$.

$$G(k_1,k_2) = \sum_{n_1=-(N_1-1)}^{N_1-1} \sum_{n_2=-(N_2-1)}^{N_2-1} g(n_1,n_2)\, e^{-j2\pi \left[\frac{n_1 k_1}{M_1} + \frac{n_2 k_2}{M_2}\right]} \tag{39}$$

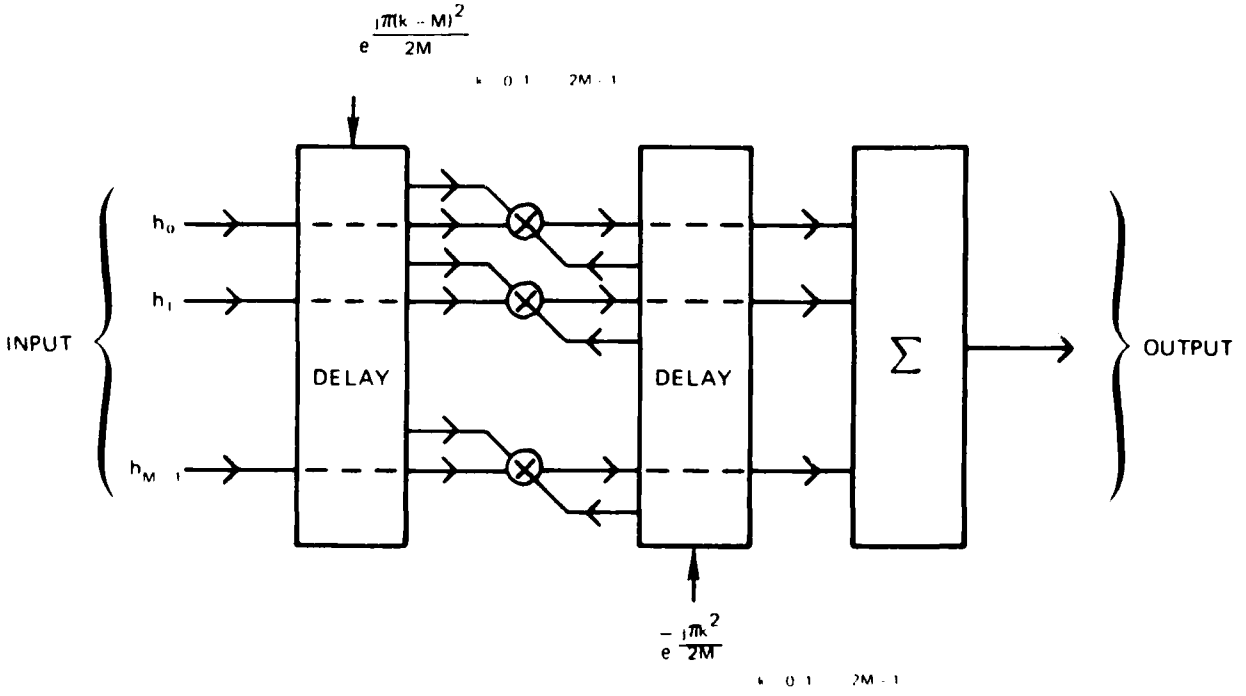


Figure 20. Parallel-Input, Serial-Output CZT Using Multi-Port Transversal Filter

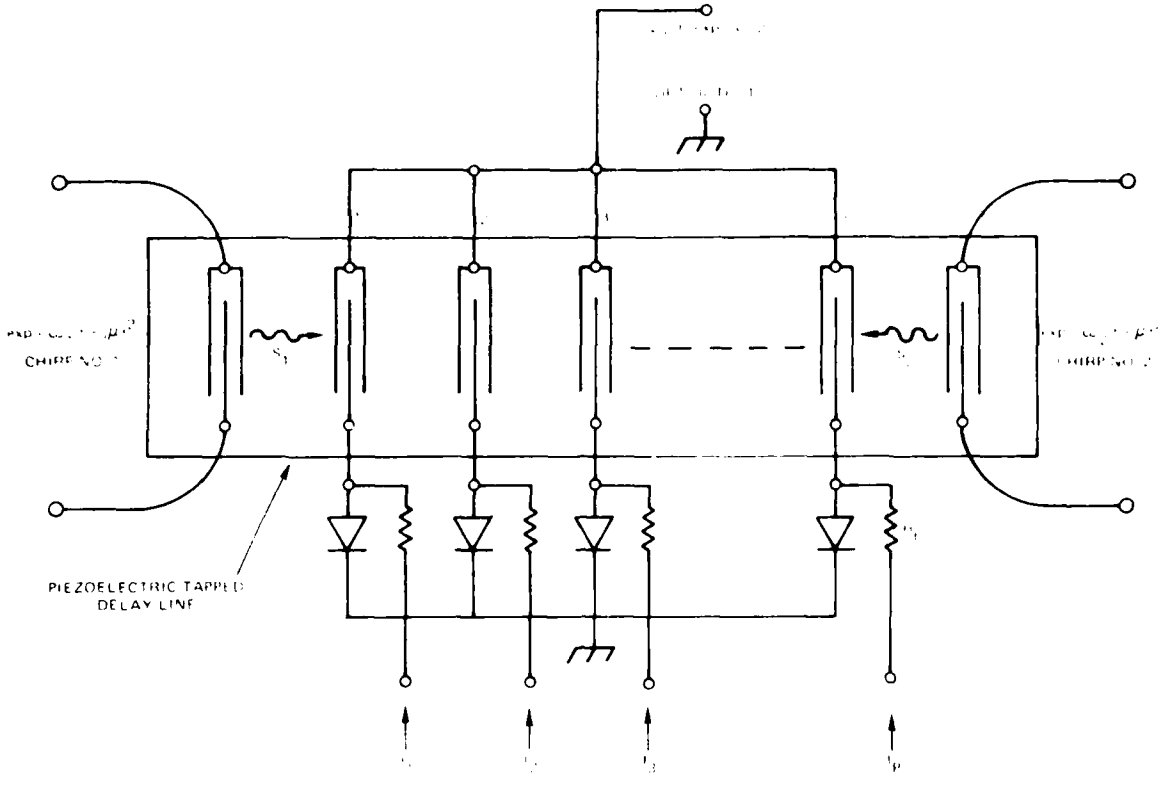


Figure 21. Diode-Correlator DFT Module

Since the indices are only defined Modulo M_1 and M_2 respectively, the summation limits in (39) are really no different from those in (22). If the scan is defined by equations (25) and (26), then the symmetry of the corresponding one-dimensional sequence is shown in equation (40).

$$f_{-n} = g(-n_1, -n_2) = g(n_1, n_2) = f_n \tag{40}$$

The toroidal scan of the extended data block, which can be obtained by repeatedly scanning points of the original data block, is illustrated in Table 1 for a block size of 2 by 3. The numbers in the table indicate the scan order, while the letters indicate the data values. Similar results have been obtained for a mixed ODCT by EDCT which permit the use of a square block size. [17]

SIMULTANEOUS COMPUTATION OF THE DFT AND THE DCT

The close relationship between the DFT and the DCT permits the use of common modules to simultaneously compute both transforms. This may be accomplished most simply when an EDCT is computed using DFT modules. The sum in the EDCT defining equation (18a) may be interpreted as a length 2N DFT of the extension of the function g by N zeros. This leads to the configuration shown in Figure 22. Alternatively, if the odd and even frequencies in the zero-filled DFT are considered separately, they may be computed using length N DFT modules as shown in Figure 23.

Table 1A. Toroidal Scan for an ODCT of a 2 by 3 Data Block

F	F	F
-7	3	-2
D	C	D
-1	-6	4
B	A	B
5	0	-5
D	C	D
-4	6	1
F	E	F
2	-3	7

$N_2 = 3$

$N_1 = 2$

Table 1B. One-Dimensional Sequence Produced by a Toroidal Scan for an ODCT of a 2 by 3 Data Block

INDEX	DATA VALUE
-7	F
-6	C
-5	B
-4	D
-3	E
-2	F
-1	D
0	A
1	D
2	F
3	E
4	D
5	B
6	C
7	F

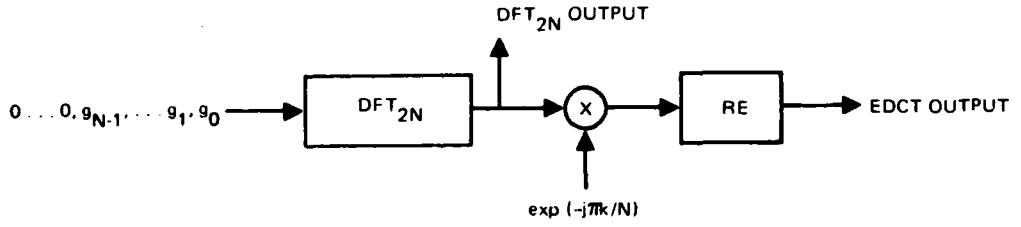


Figure 22. Computation of the DFT and EDCT Using a Single Length 2N DFT Module

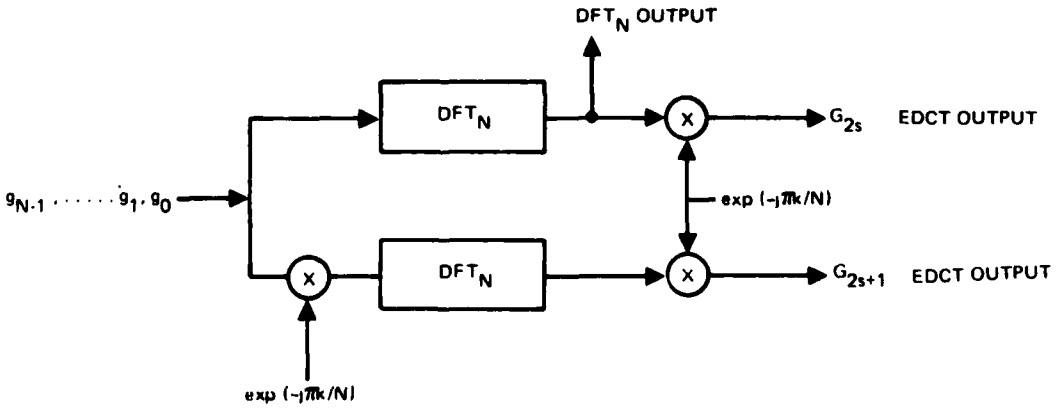


Figure 23. Computation of the DFT and EDCT Using Two Length N DFT Modules

CONCLUSION

It has been shown that a small number of multipurpose modules can provide a highly parallel structure with minimal control overhead for the computation of many linear transforms. The transforms include one- and two-dimensional discrete Fourier transforms and discrete cosine transforms. The basic modules may be chosen to be multipliers, serial access memories, and discrete chirp filters or crosscorrelators. Current acoustic surface wave and CCD technologies permit small, low power, lightweight, high speed implementations of the required modules and permit real-time solutions to highly demanding signal processing tasks including real-time video data compression, [18] spread spectrum communication, and high resolution radar signal processing.

ACKNOWLEDGMENT

This work was supported in part by Defense Advanced Research Projects Agency, Order Number 2303, Code Number 3G10, monitored by Col. H. M. Federhen.

REFERENCES

[1] Squire, W.D., Whitehouse, H.J., and Alsup, J.M. "Linear Signal Processing and Ultrasonic Transversal Filters." IEEE Transactions on Microwave Theory and Techniques, MTT-17, pp. 1020-1040, 1969.

[2] Shea, T.E., *Transmission Networks and Wave Filters*, D. Van Nostrand Co., 1929.

[3] Bluestein, L.T., "A Linear Filtering Approach to the Computation of Discrete Fourier Transform," IEEE Trans. Audio and Electroacoustics, AU 18, pp. 451-455, 1970.

[4] Rabiner, L.R., et al., "The Chirp-Z Transform Algorithm," IEEE Trans Audio and Electroacoustics, vol. AU-17, pp. 86-92, 1969.

[5] Cooley, J. W. and Tukey, John W., "An Algorithm for the Machine Calculation of Complex Fourier Series," Mathematics of Computation, Vol. 19, No. 90, pp. 297-301, 1965.

[6] Whitehouse, H. J., Speiser, J. M., and Means, R. W., "High Speed Serial Access Linear Transform Implementations," presented at the All Applications Digital Computer (AADC) Symposium, Orlando, Florida (January 23-25, 1973). Reprinted in Image Transmission via Spread Spectrum Techniques, ARPA Quarterly Technical Report, Order Number 2302 Code Number 3G 10 (February 1, 1973-June 1, 1973).

[7] Gauss, Carl Friedrich, *Disquisitiones Arithmeticae*, translated by Arthur A. Clarke, S. J. New Haven and London: Yale University Press, 1966.

[8] Rader, C., "Discrete Fourier Transforms When the Number of Data Samples is Prime," Proc. IEEE, 56, pp. 1107-1108, 1968.

[9] Byram, G. W., Alsup, J.M., Speiser, H. M., Whitehouse, H. J., "Signal Processing Device Technology," Proc. NATO Advanced Study Institute on Signal Processing, University of Technology Loughborough, U.K., IV.10.1-IV.10.13 (August 21-September 1, 1972).

[10] Alsup, J. M., Means, R.W., and Whitehouse, H.J., "Real Time Discrete Fourier Transforms Using Surface Acoustic Wave Devices," Proc. IEE International Specialist Seminar on Component Performance and Systems Application of Surface Acoustic Wave Devices, Aviemore, Scotland (September 24-28, 1973).

[11] Buss, D.D., Collins, D.R., Bailey, W.H., and Reeves, C.R., "Transversal Filtering Using Charge Transfer Devices," IEEE J. of Solid State Circuits, SC-8, pp. 138-146, 1973.

[12] Tiemann, J.J. et al., "Intra Cell Transfer Structures for Signal Processing," IEEE Transactions on Electron Devices, ED-21, pp. 300-308, 1974.

[13] Ahmed, N., Natarajan, T., and Rao, K.R., "On image processing and a discrete cosine transform," IEEE Trans on Computers, C-23, pp. 90-93, 1974.

[14] Habibi, A., "Hybrid Coding of Pictorial Data," Comm. Tech., COM-22, pp. 614-624, 1974.

[15] Cooley, James W., Lewis, Peter A.W., and Welch, Peter D., "Historical Notes on the Fast Fourier Transform," IEEE Transactions on Audio and Electroacoustics, AU-15, pp. 76-79, 1967.

[16] Reeder, T.M., Speiser, J.M., Whitehouse, H. J., "Real-Time Discrete Fourier Transforms Using a Programmable Diode-Convolver Module": to be published.

[17] Means, R.W., Whitehouse, H.J., Speiser, J.M., ARPA QR5, ARPA Order Number 2303, Code Number 3G10, Jan. 1975.

[18] Means, R.W., Whitehouse, H.J., Speiser, J.M., Television Encoding Using a Hybrid Discrete Cosine Transform and a Differential Pulse Code Modulator in Real Time." 1974 National Telecommunication Conference Record, pp. 61-74.

APPENDIX H

INVESTIGATING ADAPTIVE TRANSFORMS IN INFORMATION PROCESSING APPLICATIONS

PREFACE

This report describes work performed during the five month period from June 1, 1975 through November 27, 1975. The investigation was funded by the Naval Undersea Center, San Diego, California under contract no. N 66001-75-0-226 MJE, and was conducted at UCLA under the direction of Judea Pearl as Principal Investigator. The team engaged in this study consisted of: Judea Pearl, Massih Hamidi and Yechiam Yemini.

The continuous guidance and encouragement of Harper Whitehouse, Jeff Speiser and Robert Means of the Naval Undersea Center deserve the major credit for the accomplishments reported here.

ABSTRACT

The performances of the Discrete Cosine Transform (DCT) and the Discrete Fourier Transform (DFT) were analyzed and compared in signal processing applications. Conditions for the asymptotic optimality of the DFT and DCT were established. The superiority of the DCT over the DFT was established for Markov-1 signals. A tool for studying asymptotic behavior of transform was developed using numerical quadrature analysis.

SUMMARY OF WORK PERFORMED

Guided by Naval Undersea Center directives the emphasis of our investigation has been to analyze and compare the performance of two practical transform techniques, the Fourier (DFT) and the Cosine (DCT) transformation, in image processing applications. Our study has resulted in several new results with both theoretical and practical significance.

1. While it is well known that the Fourier coefficient of stationary finite-duration continuous signals are asymptotically uncorrelated, correlation properties of finite-dimensional, discrete Fourier Transform remained an open question. We have established the following facts:

   a. If the covariance sequence is summable the magnitude of every off-diagonal covariance element converges to zero as $N \to \infty$.

   b. If the covariance sequence is only square-summable the magnitude of the covariance elements sufficiently far from the diagonal converges to zero as $N \to \infty$.

   c. If the covariance sequence is square-summable the weak norm of the matrix containing only the off-diagonal elements converges to zero as $N \to \infty$.

   d. If the covariance sequence is summable the weak norm of the matrix containing only the off-diagonal elements converges to zero at least as fast as $\frac{1}{\sqrt{N}}$.

2. It was conjectured that the performance of the Cosine transform (DCT) is superior to that of the Fourier transform (DFT). The fact that DCT was found more compatible with the hardware configuration of the Image Processing group at NUC called for an analytical examination of this conjecture. We were successful in establishing the following results:

   a. The DCT is asymptotically equivalent to the Karhunen-Loéve transform (KLT) of Markov-1 signals and the rate of convergence is (similar to the DFT) on the order of $N^{-1/2}$.

   b. The DCT offers a better approximation to the KLT of Markov-1 signals than the DFT for all values of N and ρ.

   c. The DCT is asymptotically equivalent to the KLT of all finite-order Markov signals.

   d. The analysis of asymptotic properties of discrete transform can be simplified substantially using numerical quadrature techniques.

A detailed description of these results are contained in the following three appendices.

APPENDIX I

ON THE RESIDUAL CORRELATION OF FINITE-DIMENSIONAL DISCRETE-FOURIER-TRANSFORMS OF STATIONARY SIGNALS

Massih Hamidi

and

Judea Pearl

Published in IEEE Transactions on
Information Theory, July 1975
pp. 480-482

ABSTRACT

The covariance matrix of the Fourier coefficients of N-sampled stationary random signals is studied. Three theorems are established.

1. If the covariance sequence is summable the magnitude of every off-diagonal covariance element converges to zero as $N \to \infty$.
2. If the covariance sequence is only square-summable the magnitude of the covariance elements sufficiently far from the diagonal converges to zero as $N \to \infty$.
3. If the covariance sequence is square-summable the weak norm of the matrix containing only the off-diagonal elements converges to zero as $N \to \infty$.

The rates of convergence are also determined when the covariance sequence satisfies additional conditions.

1. INTRODUCTION

It is well known that the Fourier coefficients of stationary finite-duration continuous signals are asymptotically uncorrelated. Root and Pitcher[1] have shown that the cross-correlation between any two Fourier coefficients approaches zero as the signal duration grows infinite. Consequently, problems involving stationary stochastic processes are often treated by approximating the original processes by Fourier series with uncorrelated random coefficients.

In applications involving digital signal processing the continuous signals are sampled at a finite number N of equally spaced points in time and are treated as N-dimensional vectors. Root and Pitcher's results coupled with the computational convenience of the Fast-Fourier-Transform[2] gave rise to a number of applications whereby the finite Fourier transform of the signal vector is taken and its components are treated as uncorrelated random variables. For example, in digital transmission of pictures and voice it is a common practice to assign each Fourier component a digital code which is independent on the magnitude of the other Fourier coefficients[3].

The Fourier coefficients of a signal N-vector $\underline{x}_N = (x_0, x_1, \cdots x_{N-1})^T$ are uncorrelated if and only if the covariance matrix of $\underline{x}_N$ is a circulant matrix[4]; i.e., if $\underline{E}(x_i x_j)$ is a function only of $(i-j) \bmod N$. Since such circular symmetry is very rare in actual processes the covariance matrix of the Fourier coefficients will contain off-diagonal elements whose magnitudes affect system performance.

It is generally believed that the magnitude of each off-diagonal element and their cumulative effect both converge to zero as $N \to \infty$. In this paper we establish conditions under which these convergences take place and derive expressions for the rate at which the residual cross-correlation decays with N.

2. THEOREMS AND PROOFS

Consider a sampled sequence $\underline{x}_N = [x_0, x_1, \cdots x_{N-1}]^T$ of a stationary stochastic process with a Toeplitz correlation matrix $T_N = E\{\underline{x}_N \underline{\bar{x}}_N^T\}$ satisfying

$$(T_N)_{ij} = T_N(|i-j|) = t(|i-j|) \qquad i,j = 0, 1, \cdots N-1$$

where $\bar{x}$ designates the complex conjugate of x.

The discrete-Fourier-transform (DFT) of $\underline{x}_N$ is a vector $\underline{y}_N = [y_0, \cdots y_{N-1}]^T$ defined by $\underline{y}_N = F_N \underline{x}_N$

where

$$[F_N]_{m,n} = N^{-1/2} W_N^{mn} \qquad m,n = 0, 1, \cdots N-1$$

and

$$W_N = e^{i2\pi/N}$$

We study the behavior of the off-diagonal elements of the matrix

$$E\{\underline{y}_N \underline{\bar{y}}_N^T\} = F_N T_N F_N^{-1}$$

for large N, as well as the norm of $T_N - C_N$ where

$$C_N \triangleq \text{diag.}\left| E(|y_0|^2), E(|y_1|^2) \cdots E(|y_{N-1}|^2) \right|$$

The motivation for studying this norm lies in the fact that in many signal processing applications the performance degradation caused by the residual correlation can be upper bounded by monotonic decreasing functions of $|T_N - C_N|$[5].

Theorem 1

If $t(\ell)$ is summable then

$$\lim_{N \to \infty} E\{y_m \bar{y}_n\} = 0 \qquad m \neq n$$

Moreover, $\lim_{N \to \infty} E\{|y_m|^2\} = 2 \sum_{u=0}^{\infty} t(u) - t(0) = \sum_{u=-\infty}^{\infty} t(u)$

Proof: By definition

$$E\{y_m \bar{y}_n\} = N^{-1} \sum_{\ell=0}^{N-1} \sum_{r=0}^{N-1} W_N^{m\ell} \left[T_N\right]_{\ell r} W_N^{-nr} = N^{-1} \sum_{\ell=0}^{N-1} W_N^{\ell(m-n)} \sum_{u=\ell-(N-1)}^{\ell} W_N^{nu} t(|u|)$$

In order to obtain an expression with only single summation we use Abel's Transformation Formula for Partial Summation[6]:

$$\sum_{k=m}^{n} a_k b_k = \sum_{k=m}^{n-1} A_k (b_k - b_{k+1}) + A_n b_n - A_{m-1} b_m$$

where

$$A_k = \sum_{j=0}^{k} a_j \qquad \text{for } k \geq 0 \text{ and } A_{-1} = 0$$

Letting $a_\ell = W_N^{\ell(m-n)}$ and $b_\ell = \sum_{u=\ell-(N-1)} W_N^{nu} t(u)$ we obtain

$$N\, E\{y_m \bar{y}_n\} = \sum_{\ell=0}^{N-1} \left(\sum_{k=0}^{\ell-1} W_N^{k(m-n)} \right) W_N^{n\ell} \left[t(N-\ell) - t(\ell)\right] \qquad (1)$$

for $m \neq n$, and:

$$N\, E\{|y_m|^2\} = \sum_{\ell=0}^{N-1} W_N^{m\ell} [t(N-\ell) - t(\ell)] + N \sum_{u=0}^{N-1} W_N^{nu}\, t(u) \qquad (2)$$

From equation (1):

$$\left| E\{y_m \bar{y}_n\} \right| \leq N^{-1} \left| \sum_{\ell=0}^{N-1} \left(\sum_{k=0}^{\ell-1} W_N^{k(m-n)} \right) W_N^{n\ell}\, t(\ell) \right| + N^{-1} \left| \sum_{\ell=0}^{N-1} \left(\sum_{k=0}^{\ell-1} W_N^{k(m-n)} \right) W_N^{n\ell} t(N-\ell) \right|$$

The first term can be bounded by $N^{-1} \sum_{\ell=0}^{N-1} \ell\, |t(\ell)|$

For the second term we use (for $m \neq n$)

$$\sum_{k=0}^{N-1} W_N^{k(m-n)} = 0 = \sum_{k=0}^{N-u-1} W_N^{k(m-n)} + \sum_{k=N-u}^{N-1} W_N^{k(m-n)}$$

and obtain

$$N^{-1} \left| \sum_{\ell=0}^{N-1} \left(\sum_{k=0}^{\ell-1} W_N^{k(m-n)} \right) W_N^{n\ell} t(N-\ell) \right| \leq N^{-1} \sum_{u=1}^{N} u\, |t(u)|$$

Hence:

$$\left|E\{y_m \bar{y}_n\}\right| \leq 2N^{-1} \sum_{\ell=0}^{N} \ell |t(\ell)|$$

Since $t(\ell)$ is summable; given $\varepsilon > 0$, there exists an integer M such that

$$\sum_{\ell=M}^{\infty} |t(\ell)| < \frac{\varepsilon}{4}$$

Thus:

$$\left|E\{y_m \bar{y}_n\}\right| \leq 2\left[N^{-1} \sum_{\ell=0}^{M-1} \ell |t(\ell)| + \sum_{\ell=M}^{N} |t(\ell)|\right]$$

Now choosing N large enough to have $N^{-1} \sum_{\ell=0}^{M-1} \ell |t(\ell)| \leq \frac{\varepsilon}{4}$ establishes the first part of our proof.

From equation (2), we obtain, after some computation

$$E\{|y_m|^2\} = \sum_{u=0}^{N-1} W_N^{mu}\, t(u) + \sum_{u=1}^{N} W_N^{-mu} t(u) - N^{-1} \sum_{u=1}^{N-1} u\left(W_N^{mu} + W_N^{-mu}\right) t(u)$$

taking the limit of each term, we have:

$$\lim_{N \to \infty} N^{-1} \left|\sum_{u=0}^{N-1} u\, W_N^{\pm mu}\, t(u)\right| \leq \lim_{N \to \infty} N^{-1} \sum_{u=0}^{N-1} u|t(u)| = 0$$

also, it can be shown that

$$\lim_{N \to \infty} \sum_{u=0}^{N} W_N^{\pm mu}\, t(u) = \sum_{u=0}^{\infty} t(u) \qquad (3)$$

Hence:

$$\lim_{N \to \infty} E\{|y_m|^2\} = 2 \sum_{u=0}^{\infty} t(u) - t(0) = \sum_{u=-\infty}^{\infty} t(u)$$

Q.E.D.

To show that (3) is valid notice that $t(u)$ being summable, given any $\varepsilon > 0$ there exists an integer K such that

$$\left| \sum_{u=K}^{\infty} t(u) \right| < \frac{\varepsilon}{2} \quad \text{and} \quad \sum_{u=K}^{\infty} |t(u)| < \frac{\varepsilon}{2}$$

$$\left| \sum_{u=0}^{\infty} t(u) - \lim_{N \to \infty} \sum_{u=0}^{N} W_N^{\pm mu}\, t(u) \right|$$

$$\leq \left| \sum_{u=0}^{\infty} t(u) - \sum_{u=0}^{K-1} t(u) \right| + \left| \lim_{N \to \infty} \sum_{u=K}^{N} W_N^{\pm mu}\, t(u) \right|$$

$$\leq \left| \sum_{u=K}^{\infty} t(u) \right| + \lim_{N \to \infty} \sum_{u=K}^{N} |t(u)| < \varepsilon$$

Theorem 2

If $t(\ell)$ is square summable, $\lim_{N \to \infty} E\{y_m \bar{y}_n\} = 0$ for all elements such that $N^{-1} |m-n| \geq \varepsilon > 0$.

Proof:

From equation (1):

$$\left| N\, E\{y_m \bar{y}_n\} \right| = \left| 1 - W_N^{(m-n)} \right|^{-1} \left| \sum_{\ell=0}^{N-1} \left(W_N^{n\ell} - W_N^{m\ell} \right) [t(N-\ell) - t(\ell)] \right|$$

Applying the Cauchy-Schwarz inequality:

$$\left| E\{y_m y_n\} \right| \leq N^{-1} \left| 1 - W_N^{(m-n)} \right|^{-1} \left[\sum_{\ell=0}^{N-1} \left| W_N^{n\ell} - W_N^{m\ell} \right|^2 \right]^{1/2} \left[\sum_{\ell=0}^{N-1} |t(N-\ell) - t(\ell)|^2 \right]^{1/2}$$

$$\leq 2\sqrt{2}\; N^{-1/2} \left| 1 - e^{i(m-n)2\pi/N} \right|^{-1} \left[\sum_{\ell=0}^{N} |t(\ell)|^2 \right]^{1/2}$$

Hence, whenever $N^{-1}|m-n| \geq \varepsilon > 0$ $\lim_{N \to \infty} E\{y_m \bar{y}_n\} = 0$

Q.E.D.

Theorem 3

$$\lim_{N \to \infty} |T_N - C_N| = \left(N^{-1} \sum_{\substack{m,n=0 \\ m \neq n}}^{N-1} |E\{y_m \bar{y}_n\}|^2 \right)^{1/2} = 0 \text{ , if } t(u) \text{ is square}$$

summable.

Proof:

From equation (1) we obtain with some computations[5]

$$N^2 \sum_{\substack{m,n=0 \\ m \neq n}}^{N-1} |E\{y_m \bar{y}_n\}|^2 = N \sum_{\ell=0}^{N-1} (N\ell - \ell^2)|t(N-\ell) - t(\ell)|^2$$

Hence

$$|T_N - C_N|^2 = N^{-1} \sum_{\substack{m,n=0 \\ m \neq n}}^{N-1} |E\{y_m \bar{y}_n\}|^2 = N^{-2} \sum_{\ell=0}^{N-1} (N\ell - \ell^2)|t(N-\ell) - t(\ell)|^2$$

$$\leq 4N^{-2} \sum_{\ell=0}^{N-1} (N\ell - \ell^2)|t(\ell)|^2$$

Let $\varepsilon > 0$, arbitrary, be given. Since $t(\ell)$ is square summable there exists an integer P such that $\sum_{\ell=P}^{\infty} |t(\ell)|^2 < \frac{\varepsilon}{8}$. Thus

$$|T_N - C_N|^2 \leq 4\left[N^{-2} \sum_{\ell=0}^{P-1} (N\ell - \ell^2)|t(\ell)|^2 + \sum_{\ell=P}^{\infty} |t(\ell)|^2 \right]$$

Choosing N large enough to have $N^{-2} \sum_{\ell=0}^{P-1} (N\ell - \ell^2)|t(\ell)|^2 \leq \frac{\varepsilon}{8}$, yields

$$|T_N - C_N|^2 < \varepsilon$$

Q.E.D.

The convergence rates of both the off-diagonal elements and the norm of $T_N - C_N$ can be obtained under certain conditions from the proofs of theorems

1-3. These will be listed in the following three corollaries:

Corollary 1 - If $\ell t(\ell)$ is summable then $E\{y_m \bar{y}_n\} = o(N)$ for $m \neq n$.

Corollary 2 - If $t(\ell)$ is square summable then $E\{y_m \bar{y}_n\} = o(N^{1/2})$ for all $|m-n| = O(N)$

Corollary 3 - If $\ell^{1/2}\, t(\ell)$ is square summable then $|T_N - C_N| = o(N^{1/2})$.

Note that these conditions on $t(\ell)$ are satisfied for most processes encountered in practice, e.g., finite order Markoff or moving-average processes.

3. CONCLUSIONS

The theorems and corollaries established in this paper constitute a generalization of Root and Pitcher's results to the case of discrete-time signals and provide guidelines for selecting the proper block length N in Fourier-signal-processing applications.

REFERENCES

1. Root, W. L., and Pitcher, T. S., "On the Fourier-Series Expansion of Random Functions," Annals of Mathematical Statistics, 26(2):313-318, June, 1955.
2. Andrews, H. C., and Caspari, K., "A Generalized Technique for Spectral Analysis," IEEE Transactions on Computing, Vol. C-19, pp. 16-25, January, 1970.
3. Habibi, A., and Wintz, P. A., "Image Coding by Linear Transformation and Block Quantization," IEEE Transactions on Communication Technology, Vol. COM-19, pp. 50-62, February, 1971.
4. Gray, R. M., "Toëplitz and Circular Matrices: A Review," Stanford University Report SU-SEL-71-032, June, 1971
5. Pearl, J., "On Coding and Filtering Stationary Signals by Discrete Fourier Transforms," IEEE Transactions on Information Theory, Vol. IT-19, No. 2, pp. 229-232, March, 1973.
6. Apostol, T. M., Calculus, Vol. 1, Xerox College Publication, 2nd Ed., p. 407, 1967.

APPENDIX II

COMPARISON OF THE COSINE AND FOURIER TRANSFORMS OF MARKOV-1 SIGNALS

Massih Hamidi

and

Judea Pearl

Submitted for publication in IEEE Transactions on Acoustics, Speech, and Signal Processing

I. INTRODUCTION

In a recent paper Ahmed, et al.[1] propose a new transform called Discrete Cosine Transform (DCT) and, based on empirical evidence, conjecture that its performance is closer to the optimal Karhunen-Lóeve transform (KLT) than the other commonly used transforms (i.e., Discrete Fourier, Walsh-Hadamard, Haar). Means, et al.[2] actually used the DCT for encoding TV pictures in real time.

Pearl showed[3] that for a signal statistic characterized by a covariance matrix T, $|T - T_U|^2$ (to be defined later) constitutes a measure of performance for a transform U, in the sense that the error bounds (in coding and filtering) are increasing functions of $|T - T_U|^2$.

The purpose of this investigation is to determine the relation between $|T - T_C|^2$ (the norm obtained using the discrete cosine transform) and $|T - T_F|^2$ (the norm obtained using the discrete Fourier transform (DFT)), thus testing the conjecture of Ahmed and his collaborators.

II. DEFINITIONS AND NOMENCLATURE

Let T be a Töeplitz matrix and U an orthogonal transform. Let $T' = UTU^{-1}$ be the representation of T in the new basis, and $T'_U = \text{diag}\,(T'_{11}, T'_{22}, \cdots, T'_{ii}, \cdots, T'_{NN})$. We define T_U to be the representation of T'_U in the first basis, i.e.,

$$T_U = U^{-1} T'_U U$$

and $|T - T_U|^2$ the Hilbert-Schmidt norm of $T - T_U$, i.e.,

$$|T - T_U|^2 = \frac{1}{M}\left[\sum_{m,n=0}^{M-1} |(T - T_U)_{mn}|^2\right]$$

The cosine transform representation of a Töeplitz matrix T is given by CTC^{-1} where C is a MxM matrix defined by

$$\begin{cases} C_{oj} = \frac{2}{M}\frac{1}{\sqrt{2}} & j = 0, 1, \cdots, M-1 \\ C_{kj} = \frac{2}{M}\cos\frac{k}{2M}(2j+1) & k = 1, 2, \cdots, M-1 \\ & j = 0, 1, \cdots, M-1 \end{cases}$$

A simple algebraic manipulation shows:

$$CC^* = C^*C = \frac{2}{M} I$$

(where X^* indicates the complex conjugate transpose of X). Hence:

$$C^{-1} = \frac{M}{2} C^*$$

In contrast, the DFT is defined by a unitary matrix F where

$$F_{kj} = M^{-1/2} \exp\left(i \frac{2\pi}{M} kj\right) \qquad k,j = 0, 1, \cdots, M-1 .$$

III. COMPARISON OF $|T - T_C|^2$ AND $|T - T_F|^2$

For any orthogonal matrix U (e.g. $U = C$ or $U = F$) we have

$$|T - T_U|^2 = |T' - T'_U|^2 = |T'|^2 - \frac{1}{M}\sum_{m=o}^{M-1} |(UTU^{-1})_{mm}|^2$$

$$= |T|^2 - \frac{1}{M}\sum_{m=o}^{M-1} |(UTU^{-1})_{mm}|^2$$

i.e.: The higher the norm of the diagonal vector of the transformed matrix, the lower $|T - T_U|$ and the better the transform. Hence, to compare $|T - T_C|^2$ and $|T - T_F|^2$ it suffices to compare

$$\sum_{m=o}^{M-1} |(CTC^{-1})_{mm}|^2 \quad \text{and} \quad \sum_{m=o}^{M-1} |(FTF^{-1})_{mm}|^2 .$$

We consider matrices of the form

$$T = \begin{pmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{M-1} \\ \rho & 1 & & \cdots & \rho^{M-2} \\ \vdots & & \ddots & & \vdots \\ \rho^{M-1} & \cdots & & & 1 \end{pmatrix}$$

which represent covariance matrices of Markov-1 signals, with $0 \leq \rho \leq 1$ being the covariance coefficient between adjacent samples.

Clearly, for $\rho = 0$ and $\rho = 1$ the cosine and Fourier transforms are equivalent, since T is diagonal in both representations. For an intermediate value of ρ we obtained:

$$(CTC^{-1})_{oo} = \frac{1+\rho}{1-\rho} - \frac{2}{M}\frac{\rho(1-\rho^M)}{(1-\rho)^2}$$

and

$$(CTC^{-1})_{mm} = \frac{1}{2}\left[\frac{e^{i\alpha}+\rho}{e^{i\alpha}-\rho} + \frac{e^{-i\alpha}+\rho}{e^{-i\alpha}+\rho}\right] - \frac{\rho(1-(-1)^m\rho^M)}{M}\left[\frac{e^{-i\frac{\alpha}{2}}}{e^{-i\alpha}-\rho} + \frac{e^{i\frac{\alpha}{2}}}{e^{i\alpha}+\rho}\right]^2$$

for $m \neq 0$, where $\alpha = m\pi/\mu M$.

An elementary (but tedious) computation, leads to:

$$\sum_{m=o}^{M-1} |(CTC^{-1})_{mm}|^2 = \frac{M(1+\rho^2)}{1-\rho^2} - \frac{4\rho^2}{(1-\rho^2)^2} + \frac{2(1-\rho^{2M})\rho^2}{M(1-\rho^2)^3}[3(1+\rho^2)+4\rho] - \frac{4\rho^2(1-\rho^M)^2}{M^2(1-\rho)^4}$$

for $M = 2k$, $k > 1$ (i.e. M even ≥ 4).

Combined with

$$M|T|^2 = M\frac{1+\rho^2}{1-\rho^2} - \frac{2\rho^2(1-\rho^{2M})}{(1-\rho^2)^2}$$

we finally obtain the desired norm:

$$M|T-T_c|^2 = \frac{2\rho^2(1+\rho^{2M})}{(1-\rho^2)^2} - \frac{2\rho^2(1-\rho^{2M})}{M(1-\rho^2)^3}(3+4\rho+3\rho^2) + \frac{4\rho^2(1-\rho^M)^2}{M^2(1-\rho)^4} .$$

Note that

$$\lim_{M\to\infty} |T-T_c| = 0$$

implying that the DCT is asymptotically equivalent[4] to the KLT of Markov-1 processes. Moreover, since for large M and $\rho \neq 1$ we have

$$|T-T_c| = \sqrt{2}\,\frac{\rho}{1-\rho^2}\,O(M^{-1/2})$$

we conclude that the degradation in performance in filtering and coding[3] vanishes like $M^{-1/2}$.

In order to calculate $|T - T_F|^2$ recall[3] that for $T_{ij} = t(|i-j|)$ we have

$$(T - T_F)_{ij} = \frac{|i-j|}{M}\,[t(|i-j|) - t(M - |i-j|)]$$

and substituting $t(|i-j|) = \rho^{|i-j|}$ we obtain:

$$M|T - T_F|^2 = \frac{2\rho^2(1+\rho^{2M})}{(1-\rho^2)^2} - \frac{2(1+\rho^2)\rho^2(1-\rho^{2M})}{M(1-\rho^2)^3} - \frac{\rho^M(M^2-1)}{3}$$

It shows that the asymptotic behavior of $|T - T_F|$ for large M is identical to that of $|T - T_c|$. Thus, the performance difference between the DCT and the DFT must vanish like M^{-1} . Indeed, for large M one obtains the positive difference

$$|T - T_F|^2 - |T - T_c|^2 \simeq \frac{4\rho^2}{M^2(1-\rho^2)(1+\rho)^2} \qquad \rho < 1$$

indicating that the cosine transform is closer to optimal than the Fourier transform over the entire range of $0 < \rho < 1$.

For moderate values of M we should examine the expressions for $|T - T_F|^2$ and $|T - T_C|^2$ over the range $0 \leq \rho \leq 1$. The two are plotted, in a normalized form, in Figure 1. We chose $|T - I|^2$ as a common normalizing factor, where I is the identity matrix, and so

$$|T - I|^2 = \frac{2\rho^2}{M(1-\rho^2)^2} [M-1 - M\rho^2 + \rho^{2M}] .$$

It measures the degree of cross-correlation contained in the unprocessed signal, and therefore the maximum amount of decorrelation that can be accomplished by any transform (i.e. the KLT). The ratio

$$\frac{|T - T_U|^2}{|T - I|^2}$$

represents the fractional correlation left 'undone' by a transformation U.

Figure 1 shows that for $M = 8, 16, 64$, and for the entire range of $0 < \rho < 1$, $|T - T_F|^2$ is higher than $|T - T_C|^2$. The difference between the two are quite noticeable, occasionally reading a ratio of 2 : 1.

CONCLUSIONS

We established that the DCT is asymptotically equivalent to the KLT of Markov-1 signals and demonstrated that the rate of convergence is on the order of $M^{-1/2}$. $|T - T_C|^2$ is shown to be smaller than $|T - T_F|^2$ for all values of M and , i.e., the Discrete Cosine Transform offers a better approximation to the Karhunen-Lóeve transform of Markov-1 signals than the Discrete Fourier Transform.

REFERENCES

[1] Ahmed, N., T. Natarajan, and K. R. Rao, "On Image Processing and a Discrete Cosine Transform," IEEE Transactions on Computers, C-23, pp. 90-103, January, 1974

[2] Means, Robert W., Harper J. Whitehouse, and Jeffery M.Speiser, "Television Encoding Using A Hybrid Discrete Cosine Transform and A Differential Pulse Code Modulator in Real Time," Proceedings of the IEEE National Telecommunication Conference, San Diego, California, December, 1974

[3] Pearl, J. "On Coding and Filtering Stationary Signals by Discrete Fourier Transforms," IEEE Transactions On Information Theory, Vol. IT-19, No. 2, pp. 224-232, March, 1973

[4] Pearl, J., "Asymptotic Equivalence of Spectral Representations," UCLA-PAPER-ENG-0174, May, 1973, to be published in IEEE Transactions On Acoustics, Speech, and Signal Processing, December, 1975

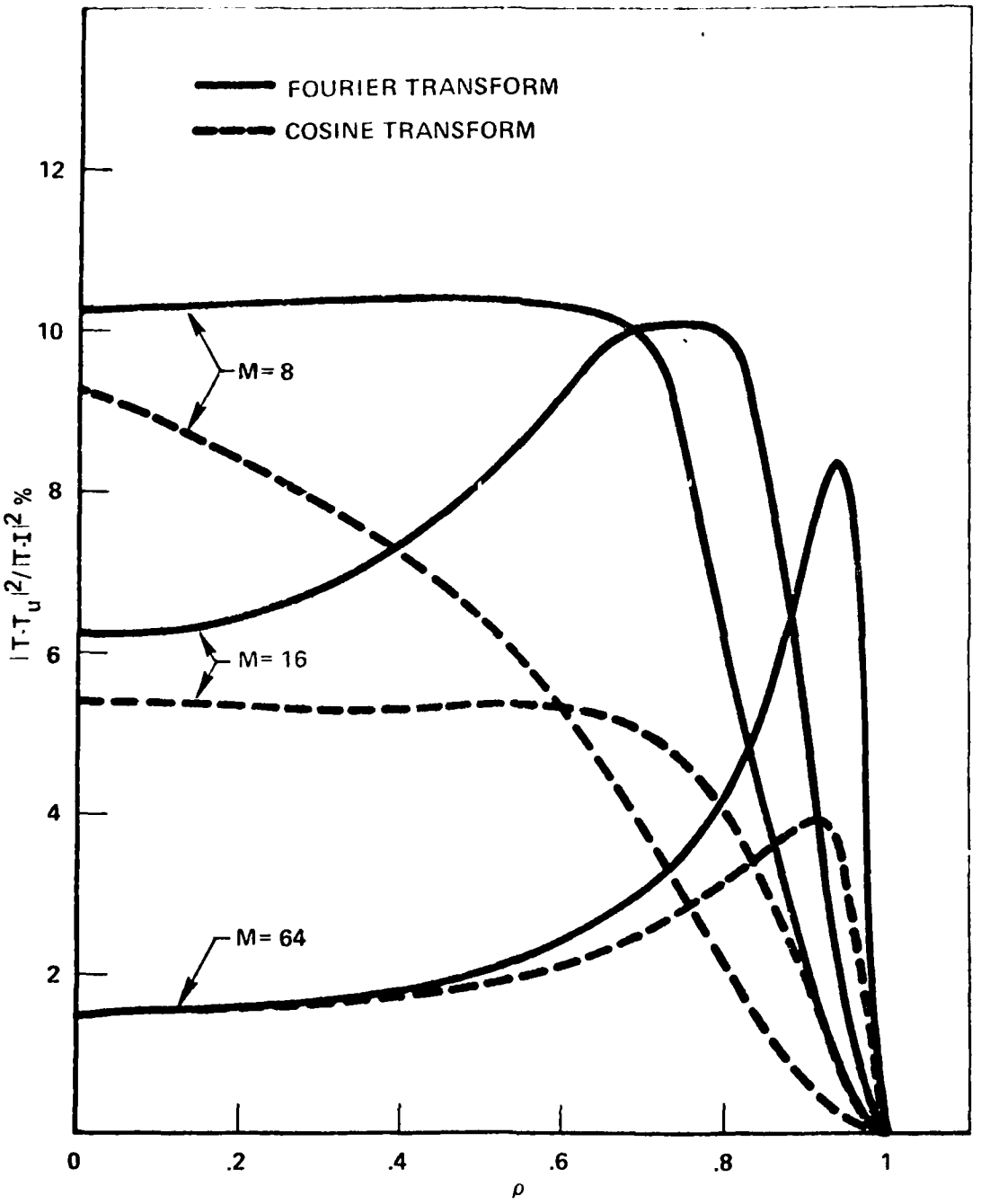


Figure 1. Normalized Correlation Measures for FOURIER (Solid-Lines) and COSINE (Broken Lines) TRANSFORMS.

APPENDIX III

ASYMPTOTIC PROPERTIES OF DISCRETE UNITARY TRANSFORMS

Yechiam Yemini

and

Judea Pearl

Submitted for publication in IEEE Transactions on Acoustics, Speech, and Signal Processing

I. INTRODUCTION

Discrete unitary transforms are used extensively in digital signal processing. Both voice and images were successfully encoded and filtered using Fourier, Cosine, Walsh, Haar, and Karhunen-Loéve expansions[1-5]. All unitary transformations are information preserving and no bandwidth reduction results from the application of the transform to the signal. Instead, its beneficial effect lies in redistributing the variance associated with each signal sample into almost uncorrelated variables, thus permitting the transform coefficients to be processed (e.g., coded or filtered) individually[6]. When the signal statistic is known one can find an optimum transformation, the Karhunen-Loéve transform (KLT), which totally decorrelates the transform coefficients and results in an optimal performance.

Real-time implementation of the Karhunen-Loéve expansion suffers from three major drawbacks: 1) A large amount of sampled-data and computational effort is required for estimating the direction cosines of the Karhunen-Loéve basis vectors. 2) The task of projecting an N-component incoming signal vector into its principal coordinates requires N^2 computer multiplications. 3) A new NxN matrix must be calculated for each change in the statistical properties of the environment.

The advantages of other transform techniques (e.g., Fourier, Cosine, Walsh) lies in the fact that they possess fast computation algorithms and that they employ a fixed set of unitary matrices independent of the signal statistics. Their performances, however,

should be judged by the degree to which each transform approximates the Karhunen-Loéve expansion in decorrelating the signals.

A good measure of the degree of correlation still remaining after the application of a specific transform is given by the norm of the matrix containing the off-diagonal covariance elements of the transformed coefficients. This norm was shown[7] to control the performance degradation resulting from residual correlation in both coding and filtering. When this norm is small one is assured that the performance degradation will remain below a tolerated level and therefore, the behavior of this norm for large N governs the proper selection of the signal dimension N. When the norm generated by a specific transform vanishes at large N we say that that transform is asymptotically equivalent to the KLT. The rate at which the norm vanishes, however, determines quality ranking among several contending transforms.

Consider a continuous signal which is sampled at N time points to give an N-dimensional vector $\underline{x}$. Let C_N be an N-dimensional unitary transform matrix and $\underline{y} \triangleq C_N \underline{x}$ the transformed signal. If $\underline{x}$ is considered a random vector with an autocovariance matrix T then the transformed autocovariance matrix is $C_N T C_N^*$ (C_N^* is the adjoint matrix of C_N). The norm determining the degree of decorrelation achieved by C_N is given by

$$\frac{1}{N} \sum_{i \neq j} \left| C_N T C_N^* \right|_{ij}^2 .$$

Now consider a sequence of such signal vectors with increasing dimension and the corresponding sequence of autocovariance matrices

$\{T_N\}_{N=1}^{\infty}$ [N will be called the block size]. We may transform each T_N using a transform of corresponding dimension C_N taken from a transform sequence $\{C_N\}_{N=1}^{\infty}$. We want to examine the degree of diagonalization of the transformed sequence $\{C_N\ T_N\ C_N^*\}_{N=1}^{\infty}$ as the block size grows to infinity.

If the signal covariance matrices were known, it would be possible to compute (numerically) the value of $\frac{1}{N}\sum_{i\neq j}\left|C_N\ T_N\ C_N^*\right|_{ij}^2$ for each N, observe its behavior or verify asymptotic equivalence. In most cases, however, a system of transforms must be decided upon with only partial knowledge of the input statistics. Techniques using the discrete-Fourier transform (DFT), or discrete cosine transform (DCT)[7], for instance, are often expected to process stationary signals of arbitrary statistics. Analytical techniques must be employed for examining asymptotic behavior of $C_N\ T_N\ C_N^*$ over an entire class of covariance matrices T_N. A basic difficulty impeding such analysis is that each element of $C_N\ T_N\ C_N^*$ consists of sums of components, where both the component values and the size of the sum are changing with N. The expressions received for those sums are formidable, which make it almost impossible to derive analytic results regarding the asymptotic behavior of a given transform.

This paper attempts to reduce this difficulty be deriving certain limit behavior of the elements of the transformed matrices. An analytical framework is developed which relates the problem at hand to known theories of analysis, thus making it possible to utilize known limit theorems of classical analysis to derive limit behavior of transforms.

2. BASIC DEFINITIONS

Let $\mathcal{S}_N$ be the space of symmetric and real N-dimensional matrices. On $\mathcal{S}_N$ we use two norms, the weak norm and the strong norm. Let $A \in \mathcal{S}_N$ be a matrix with eigenvalues $\{\lambda_i\}_{i=0}^{N-1}$. The <u>weak norm</u> of A is defined by:

$$|A|_N^2 \triangleq \frac{1}{N} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |a_{ij}|^2 = \frac{1}{N} \sum_{i=0}^{N-1} |\lambda_i|^2$$

The <u>strong norm</u> of A is:

$$\|A\|_N^2 \triangleq \sup\{ \langle Au, u \rangle : \|u\| = 1 \; u \in \mathbb{R}^N \} = \max_{0 \le i \le N-1} |\lambda_i|$$

[where $\langle , \rangle$ denotes an inner product on $\mathbb{R}^N$]

Both norms are invariant under unitary transforms. That is, if C is unitary $|A| = |C\ A\ C^*|$ and $\|A\| = \|C\ A\ C^*\|$. Moreover, $\|A\|_N \ge |A|_N$ which is where the norms acquire their names. The weak norm of the off-diagonal portion of the transformed covariance matrix (namely, $|C\ A\ C^* - DIAG(C\ A\ C^*)|$) measures the degree of residual correlation between the transformed signal components and bounds practical performance degradation resulting from residual correlations[7].

To be able to consider sequences of matrices, such as the autocovariance matrices of signals with increasing block size, we define nets. A <u>net</u> is a strongly bounded sequence of matrices $\{A_N\}_{N=0}^{\infty}$, such that for every N, A_N is a NxN matrix. Two nets, $\alpha = \{A_N\}$ and $\beta = \{B_N\}$ are called <u>asymptotic equivalent</u> if $|A_N - B_N|_N \xrightarrow[N \to \infty]{} 0$.

A <u>matrix class</u> $\mathcal{A}$, is a collection of nets. The N-section of $\mathcal{A}$ is the collection of N-dimensional matrices which belong to

nets in $\mathcal{A}$. We denote the N-section of $\mathcal{A}$ by $\mathcal{A}_N$.

Let $\mathcal{D}$ be the matrix class of all nets of positive diagonal matrices. $\mathcal{D}$ is called the diagonal class.

Let $\gamma = \{C_N\}$ be a net of unitary transform matrices and $\alpha = \{A_N\}$ an arbitrary net. Then by $\gamma\ \alpha\ \gamma^*$ we mean the transformed net $\{C_N\ A_N\ C_N^*\}$. If $\mathcal{A}$ is a matrix class we denote by $\gamma\ \mathcal{A}\ \gamma^*$ the transformed class. $\gamma\ \mathcal{D}\ \gamma^*$ we call the γ-spectral representation of the class $\mathcal{A}_N$. $\gamma^*\ \mathcal{D}\ \gamma$ is called the class diagonal in γ. As an example, the class diagonal in the net of Fourier transforms is the class of circulant matrices.

Given a transform net γ we consider the performance of γ on a class of covariance matrices $\mathcal{T}$, called the signal class. γ performs well on $\mathcal{T}$, in the sense that it approximately diagonalizes the class $\mathcal{T}$, if every net $\gamma\ \tau\ \gamma^*$ in $\gamma\ \mathcal{T}\ \gamma^*$ is asymptotically equivalent to a diagonal net. This motivates the following definition; if $\mathcal{A}$ and $\mathcal{B}$ are matrix classes $\mathcal{A}$ is said to be an asymptotic cover of $\mathcal{B}$; if for any net $\beta \in \mathcal{B}$, there is a net $\alpha \in \mathcal{A}$, such that α and β are asymptotically equivalent. We use the notation $\mathcal{A} \underset{\rightarrow}{\supseteq} \mathcal{B}$ to denote this fact. For example, the class of circulant matrices is known to cover the class of Toëplitz matrices[7]. Similarly, the class of Markov-1 Toëplitz matrices has been shown to be covered by the class diagonal in the Cosine transform[9].

The problem of evaluating the asymptotic performance of a transform technique is formulated as follows: For a given transform net γ, does the diagonal class $\mathcal{D}$ asymptotically cover $\gamma\ \mathcal{T}\ \gamma^*$? From the definition of asymptotic equivalence of nets, and the invariance of the weak norm under unitary transformations we have:

$$\mathcal{D} \supseteq \gamma \mathcal{T} \gamma^* \text{ iff } \gamma^* \mathcal{D} \gamma \supseteq \mathcal{T}.$$

That is, our problem is to determine whether the signal class is asymptotically covered by the class diagonal in γ.

Remark: These definitions and approach is a slight modification of the one presented in (Pearl [10]).

We also use the notation $\mathcal{A} \leftrightarrow \mathcal{B}$ if both $\mathcal{A}$ covers $\mathcal{B}$ and $\mathcal{B}$ covers $\mathcal{A}$. We say then that $\mathcal{A}$ and $\mathcal{B}$ are asymptotically equivalent.

3. DISCRETE GENERATION OF TRANSFORMS AND THEIR ASYMPTOTIC PROPERTIES

3.1 Numerical Quadrature and Transform Generation

Numerical quadrature is a scheme of approximating integrals by finite sums. It is discussed in any book on numerical analysis and to a deeper extent in (Krylov [11]). Our approach is to study the asymptotic behavior of transform nets using integral approximations of finite sums. A brief summary of numerical quadrature is given in the sequel.

Let X be a real interval (usually we shall take X = [-1,1] or [0,1]). Let $d\mu(x)$ be a measure on X. Consider integrals of the form $\int_X f(x)d\mu(x)$. A numerical quadrature is a sum of the form $Q_N(f) = \sum_{i=0}^{N-1} \lambda_i^N f(x_i^N)$. $\{\lambda_i^N\}_{i=0}^{N-1}$ are called the weights of the numerical quadrature and $\{x_i^N\}_{i=0}^{N-1}$ the discretization points, N is the order of the numerical quadrature. A quadrature is exact of order N if for any polynomial P(x) of order $\leq N$, $Q_n(P) = \int_X P(x)d\mu(x)$. A quadrature converges W.R.T. a given class of functions F if for any $f \in F$ $Q_n(f) \rightarrow \int_X f(x)d\mu(x)$.

Consider the case where $d\mu(x)$ is a probability measure over

$X = [-1,1]$. To demonstrate the process of generating unitary transforms by discretization, we present the Gauss-Jacobi quadrature formula. Let $\{P_N(x)\}$ be the orthonormal polynomials W.R.T. $d\mu(x)$ obtained by the application of the Grahm-Schmidt process to $x^n, n = 0,1,\cdots N$. It is known (Szego [12]) that $P_N(x)$ is a polynomial of order N which possesses N distinct roots in $(-1,1)$; $\{x_i^N\}_{i-o}^{N-1}$. The Gauss-Jacobi theorem states that there are numbers (Christoffel Numbers) $\{\lambda_i^N\}_{i=o}^{N-1}$ such that

(1) $\lambda_i^N \geq 0$ for $i = 0 \cdots N-1$, $\sum_{i=o}^{N-1} \lambda_i^N = 1$.

(2) The Gauss-Jacobi quadrature $Q_N(f) \triangleq \sum_{i=o}^{N-1} \lambda_i^N f(x_i^N)$, is exact of order 2N-1.

(3) $Q_N(f)$ converges for any f for which $\int_X f(x)d\mu(x)$ converges.

This theorem implies the following procedure for generating orthogonal transform nets. The polynomial $P_k \cdot P_j$, $0 \leq k, j \leq w-1$ is of degree $\leq 2N-1$, thus

$$Q_N(P_k \cdot P_j) = \int_{-1}^{1} P_k(x)P_j(x)d\mu(x) = \delta_{kj},$$

therefore:

$$\sum_{i-o}^{N-1} \lambda_i^N P_k(x_i^N) P_j(x_i^N) = \delta_{kj}.$$

If we let $C_{ij}^N \triangleq \sqrt{\lambda_i^N} P_j(x_i^N)$ then $\{C^N\}_{N=1}^{\infty}$ is a unitary transform net. We call transform generated by this procedure Gauss-Jacobi Transforms.

Examples: (1) Let $X = [-1,1]$ $d\mu(x) = \frac{2dx}{\pi\sqrt{1-x^2}}$ then $P_n(x) = \cos n\pi\theta$ (where $x = \cos\theta$) are the corresponding orthonormal polynomials.

The G-J transform is formed by discretizing $P_0 \cdots P_{n-1}$ at the zeros of P_n. This gives the cosine transform as discussed in (Ahmed [8]).

(2) Let $X = [-\pi,\pi]$ $d\mu(x) = \sqrt{1-x^2}\,dx, P_n(x) = \frac{\sin(n+1)\theta}{\sin\theta}$

where $x = \cos\theta$ are the Tchebychev polynomials of the second kind and give rise to the sine transform

$$S_{ij}^N \triangleq \left[\frac{\sqrt{\lambda_i^N}}{\sin\frac{i\pi}{n+1}}\right] \sin\frac{(j+1)i\pi}{N+1} = \alpha_i \sin\frac{(j+1)i\pi}{N+1}$$

where α_i are normalization factors.

If $X = [-\pi,\pi]$ and if we consider integrals of the form $\frac{1}{2\pi i}\int_{-\pi}^{\pi} f(e^{\cdot})e^{i\theta}d\theta$ then we have similar results for the Newton-Cotes Quadrature Scheme. The discretization points in this case are equally spaced in $[0, \pi)$; $x_i^N = i \cdot \frac{2\pi}{N}$, $i = 0 \cdots N-1$ and $\{\lambda_i^N\}$ are selected so that the resulting quadrature is exact of order N-1. If we take the system $\{e^{in\theta}\}_n$ of orthonormal functions then the exactness of the quadrature scheme gives the familiar Discrete Fourier Transform.

Using arbitrary measures on the unit circle and their corresponding orthonormal trigonometric polynomials one can produce many other trigonometric transforms.

3.2 Termwise Asymptotic Behavior Of Transforms - Generalized Toëplitz Matrices

Let $X = [-1, 1]$, $d\mu(x)$ a probability measure on X, and $P_k(x)$ the corresponding othonormal polynomials. If $Q_n(f)$ is the Gauss-Jacobi quadrature scheme, then by G-J theorem, this scheme converges for every function f, such that $\int_X f(x)\,d\mu(x)$ exists. In par-

ticular, if f(x) is a bounded positive function then

$$Q_N(P_k\ f\ P_j) \xrightarrow[N \to \infty]{} \int_X P_k(x)P_j(x)d\mu(x)$$

If D_N = Diagonal $[f(x_o^N) \cdot \cdot \cdot f(x_{N-1}^N)]$ and C_N is the Gauss-Jacobi Transform corresponding to $\{P_k\}$ then this simply implies that

$$[C_N^*\ D_N\ C_N]_{kj} \xrightarrow[N \to \infty]{} \int_X P_k(x)P_j(x)f(x)d\mu(x)$$

That is, if we look at the diagonal net $\tau_f \triangleq \{D_N\}$ and the G-J transform net $\gamma \triangleq \{C_N\}$, then the elements of the matrices of the transformed net, $\gamma\ \tau_f\ \gamma^*$ tend asymptotically, <u>termwise</u>, to the elements of the fixed (infinite) matrix

$$[M(f)]_{kj} = \int_X P_k(x)P_j(x)f(x)d\mu(x).$$

This means that the class diagonal in γ is <u>termwise asymptotically equivalent to the class of finite sections of infinite matrices of the form M(f)</u>.

This motivates a further examination of such classes. Given an orthonormal system $\{\phi_n(x)\}_n$ on the measure space $(X, d\mu)$ consider the infinite matrix defined by

$$M_{kj}(f) \triangleq \int_{-1}^{1} f(x)\phi_k(x)\phi_j(x)d\mu(x)$$

where f is positive bounded function. With each such matrix we associate a net of finite sections of M(f) $(i,j = 0,1,\cdots N-1)$; we use m_f to denote this net. The class of all such nets is called <u>Generalized Toeplitz class associated with $\{\phi_k(x)\}$</u>

Our result above may be restated as follows:

The class diagonal in γ is termwise asymptotically equivalent to the generalized Toëplitz class associated with $\{P_k(x)\}_k$.

Example 1: The generalized Toëplitz matrices associated with $\phi_k(x) \triangleq \frac{1}{\sqrt{2\pi}} e^{i\pi kx}$ are the ordinary Toëplitz matrices since:

$$[M(f)]_{kj} = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{i(k-j)x} f(x)dx = \hat{f}(k-j)$$

($\hat{f}(k-j)$ is the k-j Fourier coefficient of f). That is, M(f) has constant diagonals.

From here on we designate the term Toëplitz matrix to matrices $T = (t_{ij})$ of the form $t_{ij} = t(|i-j|)$ which are positive definite. Such matrices are the autocovariance matrices of stationary signals.

The previous result implies that the class diagonal in the DFT (circulant matrices) is termwise asymptotic equivalent to the Toëplitz class. This parallels the result of Hamidi and Pearl[13] showing the vanishing of the off-diagonal elements of $\gamma \tau \gamma^*$, with γ being the net of DFT matrices.

Example 2: Consider again the DCT which has been shown to be the G-J transform associated with the Chebychev polynomials of the first kind $P_k(x) = \cos n\pi\theta \quad \theta = \cos^{-1}x$. We now find the generalized Toëplitz matrices associated with this orthogonal system. Let f(x) be a positive bounded function on X = [-1, 1]. Then

$$[M(f)]_{kj} = \frac{1}{2} \int_{-1}^{1} f(\theta)\cos k\pi\theta \cos j\pi\theta \, d\theta = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\theta)\cos(k-j)\theta \, d\theta$$

$$+ \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\theta)\cos(k+j)\theta \, d\theta$$

$$M(f) = \begin{vmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_1 & \alpha_0 & \alpha_1 & \alpha_2 \\ \alpha_2 & \alpha_1 & \alpha_0 & \alpha_4 \\ & & & \end{vmatrix} + \begin{vmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_1 & \alpha_2 & \alpha_3 & \\ \alpha_2 & \alpha_3 & & \\ \alpha_3 & & & \end{vmatrix}$$

Thus, a typical M(f) is the sum of a Toëplitz matrix and a Hankel matrix (both infinite), with the same parameters. Loosely stating, our theorem above states that any cosine-diagonal net tends termwise to such a generalized cosine-Toëplitz matrix. If we can prove that:

(1) The generalized cosine-Toëplitz class is an asymptotic cover of the Toëplitz class

and (2) Termwise asymptotic behavior implies ordinary asymptotic equivalence (in weak norm sense),

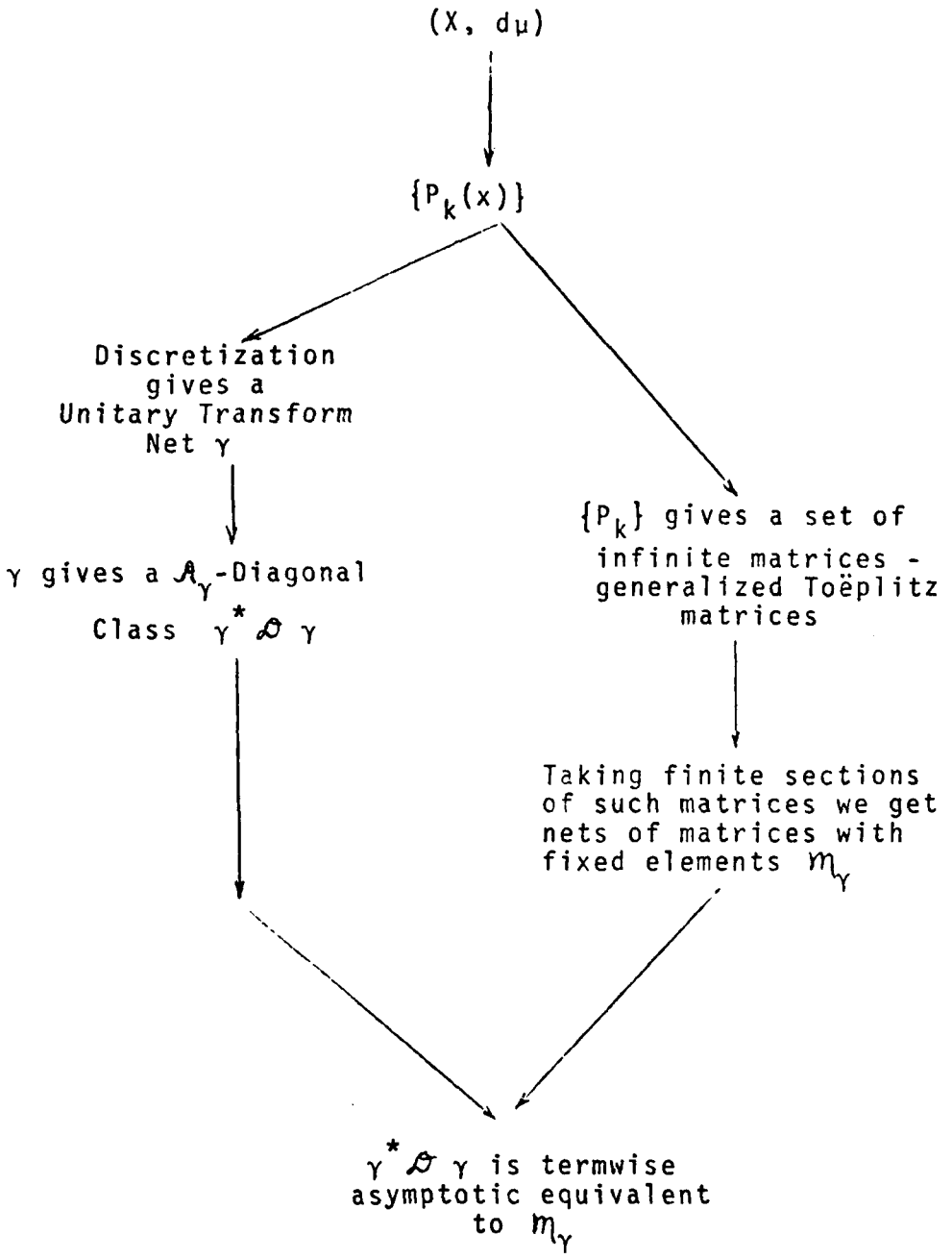
then we could show that the cosine diagonal class covers the Toëplitz class.

In summary, starting with a measure space (X $d\mu$) and an orthonormal system, we followed two paths to produce two matrix classes: $\gamma^* \mathscr{D} \gamma$, the class diagonal in a corresponding transform net, and m_γ, a generalized Toëplitz class. Termwise, these classes approximate each other asymptotically. Faced with the problem of showing $\gamma^* \mathscr{D} \gamma \rightrightarrows \tau$ we prefer to show

(1) $\gamma^* \mathscr{D} \gamma \longleftrightarrow m_\gamma$

(2) $m_\gamma \longleftrightarrow \tau$

This approach is motivated by the fact that m_γ is a more manageable matrix form than $\gamma^* \mathscr{D} \gamma$ because the elements of any net in m_γ do not vary with N. Consequently the asymptotic behavior of the nets


$(X, d\mu)$
$\{P_k(x)\}$
Discretization gives a Unitary Transform Net γ
γ gives a $\mathcal{A}_\gamma$-Diagonal Class $\gamma^* \mathcal{O} \gamma$
$\{P_k\}$ gives a set of infinite matrices - generalized Toëplitz matrices
Taking finite sections of such matrices we get nets of matrices with fixed elements m_γ
$\gamma^* \mathcal{O} \gamma$ is termwise asymptotic equivalent to m_γ

in m_γ are more transparent.

3.3 From Termwise To Weak Equivalence

Theorem 1: Let $X = [-1\ 1]$, $d\mu(x)$ a probability measure on X which has a density, $\{P_k(x)\}$ the orthogonal polynomials corresponding to $d\mu(x)$ and f a bounded positive function. Define

$$\alpha_{ij}^N \triangleq [C_N^* D_N C_N]_{ij} \qquad (C_N, D_N \text{ as in 3.2})$$

$$\alpha_{ij} \triangleq \int_X f(x)P_i(x)P_j(x)d\mu(x)$$

Then $$\lim_{N\to\infty} \frac{1}{N} \sum_{i,j=0}^{N-1} \left|\alpha_{ij}^N - \alpha_{ij}\right|^2 = 0 .$$

This means that $\gamma^* D_f \gamma$ is asymptotic equivalence to m_f.

Proof: By 3.2 we have termwise convergence, i.e. $\alpha_{ij}^N \xrightarrow[N\to\infty]{} \alpha_{ij}\ \forall\ ij$

Also, from [13] (Theorem 6.2):

$$\frac{1}{N} \sum_{i,j=0}^{N-1} \left|\alpha_{ij}^N\right|^2 \longrightarrow \int_{-1}^{1} f^2(x)d\mu(x)$$

and $$\frac{1}{N} \sum_{i,j=0}^{N-1} \left|\alpha_{ij}\right|^2 \longrightarrow \int_{-1}^{1} f^2(x)d\mu(x)$$

Now $$\frac{1}{N} \sum_{i,j=0}^{N-1} \left|\alpha_{ij}^N - \alpha_{ij}\right|^2 = \frac{1}{N} \sum_{i,j=0}^{N-1} [(\alpha_{ij}^N)^2 + \alpha_{ij}^2] - \frac{2}{N} \sum_{i,j=0}^{N-1} \alpha_{ij}^N \alpha_{ij}$$

Let $N \to \infty$ then the first sum goes to $2\int_{-1}^{1} f^2(x)d\mu(x)$.

Now $$\left|\frac{1}{N} \sum_{i,j=0}^{N-1} \alpha_{ij}^N \alpha_{ij} - \frac{1}{N} \sum_{i,j=0}^{N-1} (\alpha_{ij})^2\right| = \frac{1}{N}\left|\sum_{i,j=0}^{N-1} \alpha_{ij}^N(\alpha_{ij}^N - \alpha_{ij})\right| ,$$

$$\leq \frac{1}{N} \sum_{i,j=o}^{N-1} \alpha_{ij} \left| \alpha_{ij}^{N} - \alpha_{ij} \right|$$

and (as generalized Fourier coefficient of f) $\lim_{\substack{i \to \infty \\ j \to \infty}} \alpha_{ij} = 0$,

Let $\varepsilon > 0$ and let N be such that $\alpha_{ij} < \varepsilon$ for $i,j \geq N-1$. If we choose $N_o > N$ large enough such that $\left| \alpha_{ij}^{N} - \alpha_{ij} \right| < \varepsilon$ for $i,j \leq N-1$ we get

$$\frac{1}{N_o} \sum_{i,j=o}^{N_o-1} \alpha_{ij} \left| \alpha_{ij}^{N_o} - \alpha_{ij} \right| = \frac{1}{N_o} \sum_{i,j=o}^{N-1} \alpha_{ij} \left| \alpha_{ij}^{N_o} - \alpha_{ij} \right|$$

$$+ \frac{1}{N_o} \sum_{i,j=N-1}^{N_o-1} \alpha_{ij} \left| \alpha_{ij}^{N_o} - \alpha_{ij} \right|$$

$$\leq \varepsilon \left[\frac{1}{N_o} \sum_{i,j=o}^{N-1} \alpha_{ij} + \frac{1}{N_o} \sum_{i,j=N-1}^{N_o-1} \left| \alpha_{ij}^{N_o} - \alpha_{ij} \right| \right]$$

These two sums are bounded because

$\frac{1}{N} \sum_{i,j=o}^{N-1} \alpha_{ij}$ tends to $\int_{-1}^{1} f(x)d\mu(x)$ ([13] (Theorem 6.2)) and $\alpha_{ij}^{N_o} \longrightarrow 0$ as $i,j \to \infty$. Thus

$$\left| \frac{1}{N} \sum_{i,j=o}^{N-1} \alpha_{ij}^{N} \alpha_{ij} - \frac{1}{N} \sum_{i,j=o}^{N-1} \alpha_{ij}^{2} \right| < \varepsilon \text{ constant}$$

which proves that

$$\frac{2}{N} \sum_{i,j=o}^{N-1} \alpha_{ij}^{N} \alpha_{ij} \longrightarrow 2 \int_{-1}^{1} f^2(x)d\mu(x)$$

Thus

$$\frac{1}{N} \sum_{i,j=o}^{N-1} \left| \alpha_{ij}^{N} - \alpha_{ij} \right|^2 \xrightarrow[N \to \infty]{} 0$$

Q.E.D.

SUMMARY:

We have shown that for measures with density

$$\gamma^* \mathcal{O} \gamma \leftrightarrow \mathcal{M}_\gamma$$

This means that to find out whether $\gamma^* \mathcal{O} \gamma \supseteq \mathcal{T}$ it is enough

(a) To find the form of generalized Toëplitz matrices corresponding to $d_\mu(x)$,

(b) Compare those infinite matrices with signal class $\mathcal{T}$, and prove their equivalence.

3.4 Applications

1. We first use this method to demonstrate that the cosine-diagonal class is an asymptotic cover of a useful Toëplitz subclass. Consider the class $\mathcal{M}_\gamma$ with γ being the CDT net. As shown before, a typical net is of the form

$$\begin{vmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_1 & \alpha_0 & \alpha_1 & \alpha_2 \\ \alpha_2 & \alpha_1 & \alpha_0 & \alpha_1 \\ \alpha_3 & & & \end{vmatrix} + \begin{vmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \alpha_3 \\ \alpha_1 & \alpha_2 & \alpha_3 & \\ \alpha_2 & \alpha_3 & & \\ \alpha_3 & & & \end{vmatrix}$$

We want to show that the Toëplitz part is asymptotic equivalent to the Toëplitz plus Hankel net. This means that the Hankel net must tend (in weak norm) to zero, i.e., we want to show

$$\lim_{N \to \infty} \frac{1}{N} \left[\alpha_0^2 + 2\alpha_1^2 + 3\alpha_2^2 + \cdots N \cdot \alpha_{N-1}^2 \right] = 0$$

Clearly, if we restrict our Toëplitz class to satisfy the additional smoothness condition: $\sum_{m=0}^{\infty} m\, \alpha_m^2 < \infty$ then the limit above approaches

O like N^{-1}.

We conclude, therefore, that the class of stationary signals satisfying $\sum_{m=o}^{\infty} m\,\alpha_m^2 < \infty$ is asymptotically covered by the cosine transform. This is a substantial generalization of Hamidi and Pearl's[9] result. It implies, for instance, that the DCT is asymptotically optimal for all finite-order Markov signals. Moreover, the proof of this statement is surprisingly simple involving none of the tedious calculations used in [9] for Markov-1 processes.

2. The same procedure can be applied to polynomials on the unit circle (trigonometric polynomials) which yield the DFT by discretization. In this case the equivalence $\gamma^* \mathcal{L} \gamma \leftrightarrow \mathcal{M}_\gamma$ has a particular meaning: the left hand side are the circulant matrices and the right hand side is just the ordinary Toëplitz class. Thus we readily obtain the result that the circulant class is asymptotic equivalent to the Toëplitz class.

CONCLUSIONS

A method for studying the asymptotic behavior of spectral transformations was developed using numerical quadrature theory. Using this method, the asymptotic optimality of common unitary transforms can be tested conveniently. A practical new result obtained by this method states that the discrete cosine transform is asymptotically equivalent to the Karhunen-Loéve transform of stationary signals satisfying $\sum_{m=o}^{\infty} m\,\alpha_m^2 < \infty$ (e.g. finite order Markov processes).

REFERENCES

1. H. C. Andrews and W. K. Pratt, "Transform image coding," in *Polytech. Inst. Brooklyn Symp. Computer Processing in Communication*, Apr. 1969, pp. 63-84.

2. W. K. Pratt and H. C. Andrews, "Application of Fourier-Hadamard transformation to bandwidth compression," in *M.I.T. Symp. Picture Bandwidth Compression*, Apr. 1969.

3. A Habibi and P. A. Wintz, "Image coding by linear transformation and block quantization," *IEEE Trans. Commun. Technol.*, vol. COM-19, pp. 50-62, Feb. 1971.

4. S. J. Campanella, and G. S. Robinson, "Digital sequency decomposition of voice signals," *Proceedings of the Symposium on the Applications of Walsh Functions*, Washington, D. C., March 1970, pp. 230.

5. R. W. Means, H. J. Whitehouse, and J. M. Speiser, "Television encoding using a hybrid discrete cosine transform and a differential pulse code modulator in real time," *Proceedings of the IEEE National Telecommunication Conference*, December 2, 1974, San Diego, California.

6. J. Pearl, "Basis-restricted transformations and performance measures for spectral representations," *IEEE Trans. Inform. Theory*, vol. IT-17, pp. 751-752, November 1971.

7. J. Pearl, "On coding and filtering stationary signals by discrete Fourier transforms," *IEEE Trans. Information Theory*, vol. IT-19, No. 2, pp. 229-232, March 1973.

8. N. Ahmed, T. Natarajar, and K. R. Rao,*IEEE Trans. Comput.* vol. C-23, pp. 90-93, January 1974.

9. M. Hamidi and J. Pearl, "A comparison of Fourier and cosine transforms of Markov-1 signals," *UCLA-EAL-REPORT-7565*, November 1975. Submitted to IEEE Trans. on Acoustics, Speech and Signal Processing.

10. J. Pearl, "Asymptotic equivalence of spectral representations," To be published in IEEE Trans. on Acoustics, Speech, and Signal Processing, December 1975.

11. V. I. Krylov, *Approximate Calculation of Integrals*, New York, Macmillan, 1962.

12. G. Szego, *Orthogonal Polynomials*, New York, American Mathematical Society, 1959.

13. V. Grenander and G. Szego, "Toëplitz forms and their applications," *University of California Press*, Berkeley and Los Angeles, 1958.

14. M. Hamidi and J. Pearl, "On the residual correlation of finite-dimensional discrete Fourier transforms of stationary signals," *IEEE Trans. on Information Theory*, vol. IT-21, pp. 480-482, July 1975.

APPENDIX I

REAL TIME TELEVISION IMAGE BANDWIDTH REDUCTION USING CHARGE TRANSFER DEVICES

REAL TIME TELEVISION IMAGE BANDWIDTH REDUCTION USING CHARGE TRANSFER DEVICES*

H. J. Whitehouse
R. W. Means
E. H. Wrench

Abstract

Recent advances in analog semiconductor technology have made possible the direct sensing and processing of television images. By combining a charge transfer device (CTD) imager and a CTD transversal filter, real time image sensing and encoding have been achieved with low power integrated circuits so that digital transmission and bit rate reduction are made possible using differential pulse code modulation (DPCM). Good mean square error performance and freedom from DPCM artifacts are possible in a hybrid intraframe image encoder. The hybrid transform encoder performs a discrete cosine transform (DCT) on each line of the television image as it is scanned. This compacts the variance into low frequency coefficients and the DPCM encodes the corresponding DCT coefficients between successive lines. Computer simulation of this hybrid coding technique has shown good performance on 256 x 256 pixel images at 0.5 bits/pixel and channel bit error rates of 10^{-2}. An experimental system using a low resolution General Electric 100 x 100 charge injection device camera and a Texas Instruments bucket brigade transversal filter as part of the DCT processor has been constructed and provides good low resolution image quality at 1 bit/pixel and bit error rates of 10^{-3}. A high resolution vidicon compatible system is also being constructed.

Introduction

Unitary transforms for image encoding have been used for intraframe encoding.[1] In addition, these techniques may also be applied to interframe and multispectral encoding. However, all unitary transformations are information preserving and no bandwidth reduction results from the application of the transform to the image. Instead, the transforms redistribute the variance associated with each picture element (pixel) so that subsequent to the transform, basis restricting operations on the transform coefficients will result in bandwidth reduction. Upon reconstruction of the original image from the basis restricted transform coefficients, a degraded version of the original image can be obtained. Unfortunately, the interrelationship between the type of transform, the form of the noninvertible operation, and the type of degradation in the reconstructed image is very complicated and subjective. The universally used analytic criterion of the mean-square-error is, at present, the best compromise technique for transform comparison.

For the particular operation of basis restriction by truncation, a particularly simple interpretation of the bandwidth reduction can be made. The transforms may be viewed as a variance redistributing operation that approximately decorrelates the transform coefficients while transforming the variance associated with each picture element into the low-order coefficients of the transform. Under the assumption that each set of picture elements can be considered as a sample function from a wide sense stationary random process with correlation function $r^{|\tau|}$, there exists an optimum discrete transformation, the Karhunen-Loeve transformation, which totally decorrelates the transform coefficients and maximally compacts the variance to the low-order coefficients. All other transformations can be compared in their performance by comparing their transform coefficient decorrelation and variance compaction with this optimum transformation.

This intuitive interpretation can be made rigorous through the use of the rate-distortion criterion.[2, 3] It has been found from experience that the closer the eigenvectors of the transformation are to the eigenvectors of the optimum Karhunen-Loeve transformation the greater the variance is compacted and the more the coefficients can be truncated while maintaining a fixed rate distortion or mean-square-error.

Transform Encoding

Karhunen-Loeve Transformation

If a continuous time function of zero mean and autocorrelation function $R(\tau) = e^{-\alpha|\tau|}$ is considered to be a sample function from a wide-sense stationary random process, then this time function can be explicitly expanded by the Karhunen-Loeve expansion[4] and the resulting coefficients will be uncorrelated. For a discrete function of zero mean and autocorrelation function $R(\tau) = r^{|\tau|}$, which may be considered as a sample function from a first-order Markov process, a similar discrete Karhunen-Loeve transformation may be defined.[5] This transformation diagonalizes the covariance matrix and is optimal in the mean-square-error sense for a restricted set of basic functions that do not span the complete space.

The discrete Karhunen-Loeve expansion is given by[5] for the case $N = 2m$ as

$$G_k = \sum_{n=1}^{2m} \frac{2}{2m + \lambda_n^2} \sin\left\{\omega_n [k - (2m+1)/2] + n\pi/2\right\} g_n$$

$$k = 1, 2, \ldots, 2m \qquad (1)$$

where

$$\lambda_n^2 = \frac{1 - r^2}{1 - 2r\cos\omega_n + r^2} \qquad (2)$$

To be published in SPIE Conference Proceedings (Aug. 1975).

and ω_n are the first N positive roots of

$$\tan 2m\omega = \frac{-(1-r^2)\sin\omega}{(\cos\omega - 2r + r^2\cos\omega)} \tag{3}$$

Since the discrete Karhunen-Loeve expansion involves both the solution of a transcendental equation and the evaluation of the autocorrelation function of the data to be transformed, real time computation of this transform is quite complex. However, Habibi and Wintz[1] have shown that Karhunen-Loeve transformations calculated using approximate autocorrelation functions are satisfactory for many applications.

Discrete Fourier Transform

Since the discrete Fourier transform is asymptotic to the Karhunen-Loeve transformation[6] for the exponential covariance function and the basis vectors are picture independent, the Fourier transform represents a logical choice for real time implementation. The Fourier transform exists for all data lengths N. This is defined by

$$G_k = \sum_{n=0}^{N-1} e^{-i\pi 2nk/N}\, g_n \quad k = 0, 1, \ldots N-1 \tag{4}$$

Discrete Cosine Transform

Two different types of discrete cosine transform (DCT) are useful for reduced redundancy television image transmission. Both are obtained by extending the length N data block to have even symmetry, taking the discrete Fourier transform (DFT) of the extended data block, and saving N terms of the resulting DFT. Since the DFT of a real even sequence is a real even sequence, either DCT is its own inverse if a normalized DFT is used.

The "Odd DCT" (ODCT) extends the length N data block to length 2N - 1, with the middle point of the extended block as a center of even symmetry. The "Even DCT" (EDCT) extends the length N data block to length 2N, with a center of even symmetry located between the two points nearest the middle. For example, the odd length extension of the sequence A B C is C B A B C, and even length is C B A A B C. In both cases, the symmetrization eliminates the jumps in the periodic extension of the data block which would occur if one edge of the data block had a high value and the other edge had a low value, in effect, it performs a sort of smoothing operation with no loss of information. It will be noted that the terms "odd" and "even" in ODCT and EDCT refer only to the extended data block – in both cases the extended data block has even symmetry.

Let the data sequence be $g_0, g_1, \ldots, g_{N-1}$. The ODCT of g is defined as

$$G_k = \sum_{n=-(N-1)}^{N-1} g_n\, e^{\frac{-i2\pi nk}{2N-1}} \quad \text{for } k = 0, 1, \ldots, N-1 \tag{5}$$

where

$$g_{-n} = g_n \text{ for } n = 0, 1, \ldots N-1. \tag{6}$$

By straightforward substitution it may be shown that

$$G_k = 2\,\mathrm{Re} \sum_{n=0}^{N-1} \tilde{g}_j\, e^{\frac{-i2\pi nk}{2N-1}} \tag{7}$$

where $\tilde{g}$ is defined by equation (8).

$$\tilde{g}_n = \begin{cases} 0.5\, g_0, & n = 0 \\ g_n, & n = 1, \ldots, N-1 \end{cases} \tag{8}$$

The EDCT of g is defined by equation (9), where the extended sequence is defined by equation (10).

$$G_k = e^{\frac{-i\pi k}{2N}} \sum_{n=-N}^{N-1} g_n\, e^{\frac{-i2\pi nk}{2N}} \quad \text{for } k = 0, 1, \ldots, N-1 \tag{9}$$

$$g_{-1-n} = g_n \quad \text{for } n = 0, 1, \ldots, N-1 \tag{10}$$

If the mutually complex conjugate terms in equation (9) are combined, then equation (11) results. Equation (11) may be viewed as an alternate way of defining the EDCT.

$$G_k = 2\,\mathrm{Re}\left\{ e^{\frac{-i\pi k}{2N}} \sum_{n=0}^{N-1} g_n\, e^{\frac{-i2\pi nk}{2N}} \right\}$$

$$= 2 \sum_{n=0}^{N-1} g_n \cos\left[\frac{2\pi (n+0.5)\,k}{2N}\right] \qquad (11)$$

Ahmed[7] has investigated the use of the EDCT as a substitute for the Karhunen-Loeve transform and finds that it is superior to the Fourier transform and is comparable to the Karhunen-Loeve (K-L) in rate-distortion performance while maintaining the computation simplicity of a transform which does not depend on the picture statistics. Habibi[8] has shown by simulation that the DCT is equivalent in a mean-square-error sense to the K-L transform under basis restriction.

Hybrid Transforms

Mixed transforms can be used either for intraframe encoding or interframe encoding depending on the available memory and the type of transforms implemented. In order for a mixed transform to be competitive with one of the conventional two-dimensional transforms, it must offer either superior performance or simplicity of implementation. Of the transforms examined the odd length cosine transform is competitive in performance since it can be implemented as the real part of a CZT and since the transform samples are real and are the samples of an autocorrelation function which may then be extrapolated by well-known techniques. In simulations of transform performance, the cosine transform has been shown to closely approximate the behavior of the Karhunen-Loeve transform.

The benefits of mixed transformation implementation and minimum memory may be achieved for digital transmission by combining a one- or two-dimensional unitary transform with generalized DPCM in a hybrid system. The basis of operation of the hybrid transform is that the unitary transform decorrelates the image within its constraints of transform type, dimensionality, and block size, while the generalized DPCM removes the correlation between transform blocks. This hybrid system is particularly attractive for remote sensor application since it has been found that its performance is approximately as good as the Karhunen-Loeve transform and its implementation requires minimum memory.[8]

Implementation

Computational Modules

A linear transform on sampled data of finite extent may be viewed as the multiplication of a vector by a matrix. Multiplication by diagonal, circulant, or Toeplitz matrices may be accomplished rapidly with simple computational hardware modules. Multiplication by an N × N diagonal matrix requires only a scalar multiplier and a memory containing N values to provide serial access to the reference function. Multiplication by an N × N Toeplitz matrix corresponds to a convolution and may be performed using a transversal filter having 2N−1 taps. Multiplication by an N × N circulant is a special case of multiplication by N × N Toeplitz matrix in which the length of the transversal filter may be reduced to N taps if the data block is recirculated through the filter or reread into the filter from a buffer memory.

One-Dimensional DFT

Linear filters have been used for many years for the calculation of the power spectra of continuous signals. One of the earliest methods used a bank of wave filters to measure the spectra in fractional octave bands for telephone network equalization.[9] However, when increased resolution was required the number of filters rapidly become unmanageable. An alternative which overcame the difficulty of a large number of filters each with small time-bandwidth product was to substitute one linear fm (chirp) filter with large time-bandwidth product and to employ matched filtering. In this system the signal to be analyzed is used to single sideband (SSB) modulate a locally generated chirp signal and the composite modulated signal is filtered in a chirp delay line matched filter. Each component of the input signal spectrum shifts the locally generated chirp to a different position in the spectrum after SSB modulation and these shifted chirps then correlate with the reference signal represented as the impulse response of the matched filter at different times. Thus the output signal amplitude-time history reflects the amplitude-frequency composition of the input signal.

Bleustein[10] recognized that the discrete Fourier transform (DFT) of sampled data was amenable to a similar interpretation. In addition to just calculating the magnitude of the Fourier transform, linear filters could calculate the phase and thus all of the operations such a cross convolution and a cross correlation could be calculated. This technique came to be called the chirp-Z transform (CZT) and can be applied to other problems besides just the calculation of the DFT.[11] Prior to these developments, digital computation of the DFT had been significantly improved by the use of a special algorithm called the fast Fourier transform (FFT) which was described by Cooley and Turkey.[12] The FFT algorithm gained rapid popularity in signal processing since it allowed the calculation of the DFT to be done using significantly fewer machine operations (multiplications) than direct evaluation.

By direct inspection it is observed that, if symmetries of the function $\exp j\pi 2nm/N$ are not exploited, then the number of complex multiplications required will be N^2 corresponding to N multiplications for each frequency component evaluated. Even on high speed digital computers this can become the limiting consideration in signal processing applications. The advantage of the FFT algorithm is that for highly composite values of the DFT size N the number of multiplications is proportional to $N \log_2 N$.

Although the FFT has been successful in substantially reducing the computing time and cost of using general purpose digital computers it has several disadvantages for special purpose real time computation. At high throughput rates which are required for real time image processing the processor either must operate $\log_2 N$ times faster than the data rate or pipeline structures which use distributed

memory and $\log_2 N$ multipliers must be used. In addition, the internal arithmetic of the FFT processor must be done at increased precision in order to compensate for the multiple round off errors introduced by the successive stages in the FFT processor. Although these difficulties can be overcome, it is not always possible to arrange the computation in a form where the size of the transform is highly composite. For the above reasons and because of the difficulty of obtaining small, low power, fast analog to digital converters, linear transversal filter implementations of the chirp-Z-transform are attractive[13] rather than the previous CZT implementation which used an FFT to perform the required convolution.

The DFT may be easily reduced to the form suitable for linear filtering by the substitution

$$2nm = n^2 - (n-m)^2 + m^2 \tag{12}$$

which changes a product of variables into a difference so that

$$G_m = e^{-j\pi m^2/N} \sum_{n=0}^{N-1} e^{j\pi(n-m)^2/N} \; e^{-j\pi n^2/N} \; g_n \tag{13}$$

This form is seen to be equivalent to factoring the Fourier matrix F into the product of three matrices

$$F = DTD \tag{14}$$

where D is a diagonal matrix with elements $d_{nn} = \exp(-j\pi n^2/N)$ and T is a Toeplitz matrix with elements $t_{nm} = \exp(j\pi(n-m)^2/N)$.

The CZT algorithm is easily implemented by transversal filter techniques. In this case the DFT is computed by premultiplication by a discrete chirp, convolution with a discrete chirp, and postmultiplication by a discrete chirp. Figure 1 shows this configuration. However, it must be remembered that both the multiplications and convolutions are complex and a suitable representation of the complex numbers must be used. One representation is by real and imaginary part. Figure 2 shows the DFT organized as a CZT and implemented with parallel conputation of the real and imaginary parts. In Figure 2 the input signal is represented as $g = g_R + jg_I$ and the output signal is represented as $G = G_R + jG_I$, where it is understood that $g = g_n$ $n = 0, \ldots, N-1$ and $G = G_n$ $n = 0, \ldots, N-1$.

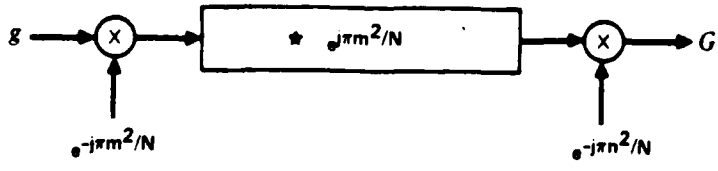
In order to determine the specific form of the transversal filters it is necessary to know the specific value of N. When N is odd the Toeplitz matrix T may be represented as a transversal filter with $2N-1$ complex taps $h_{-(N-1)}$ to h_{N-1} where $h_n = W^{-n^2/2}$, $n = -(N-1)$ to $N-1$, and $W = \exp(-j2\pi/N)$. The required convolution has been implemented with the general transversal filter shown in Figure 3.

When N is even, it can be shown that $T_{n,m} = T_{N+n,\,m}$ where the subscripts are reduced mod N. Thus T is a circulant matrix and can be implemented with a recirculating transversal filter as shown in Figure 4 where the number of complex taps is N and the tap weights are: $h_n = W^{-n^2/2}$ $n = 0, \ldots, N-1$.

Charge Coupled Devices

CCDs are sampled data analog circuits which can be fabricated by Metal Oxide Semiconductor (MOS) technology as LSI components.[14] As such they are directly compatible with other MOS circuits. Current CCD transversal filters have operated as video devices with sample rates up to 5 MHz. CCDs operate by the manipulation of injected minority carriers in potential wells under MOS capacitors and thus behave as capacitive reactances with low power dissipation. However, since the potential wells which contain the minority carriers also attract thermally generated minority carriers, there is a maximum storage time for the analog signal which depends on the dark current associated with the temperature of the silicon. Under normal conditions at room temperature, dark currents are tens of nAmps/cm^2 and storage times of hundreds of milliseconds can be achieved.

There are many ways in which unidirectional charge transfer can be achieved. The first developed was a three-phase clocking structure which is illustrated in the transversal filter of Figure 5. The three electrode CCD structure is planar, much like the SAW devices, and the direction of charge propagation is determined by the sequence of potentials applied to the three electrodes. Unfortunately, if the minority carriers are allowed to collect at the semiconductor-oxide boundary, poor charge transfer efficiency will result due to minority carriers getting caught in trapping sites. This means that the CCD will behave nonlinearly unless there is sufficient propagating charge present to fill all of the traps. By biasing the operating condition of the CCD so that about 10% of the dynamic range is used for the injection of a "fat zero," the traps are kept continuously filled and the device has over a 60 dB dynamic range. In practice, a video signal representing the signal to be processed is added to a fixed bias somewhat larger than one-half of the peak-to-peak value of the signal. Since the effective storage time of the device is long relative to the time required to execute a convolution, CCDs can be considered to be



★ Denotes either convolution or circular convolution

Figure 1. Chirp-Z Transform Implementation of the DFT

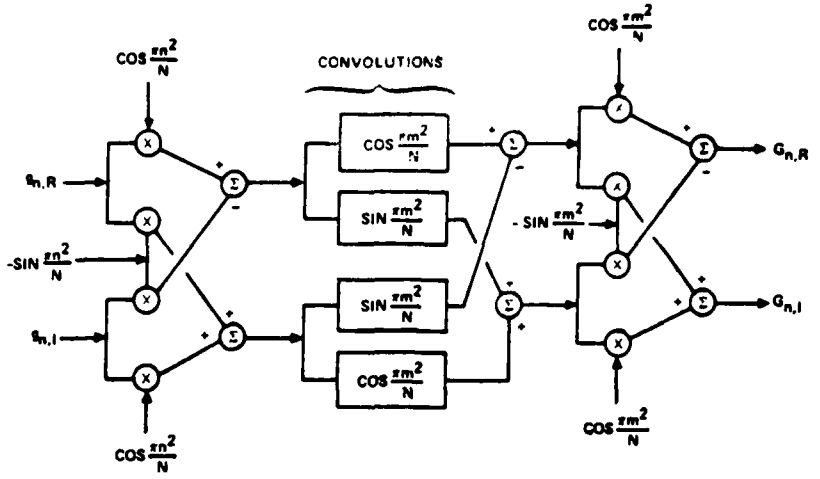


Figure 2. DFT Via CZT Algorithm with Parallel Implementation of Complex Arithmetic

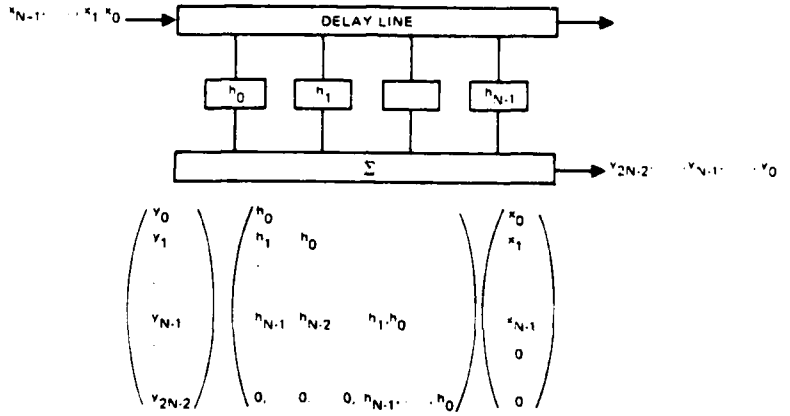


Figure 3. Transversal Filter

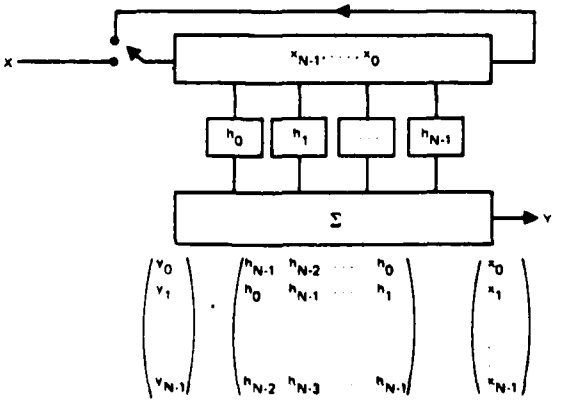


Figure 4. Circular Convolution

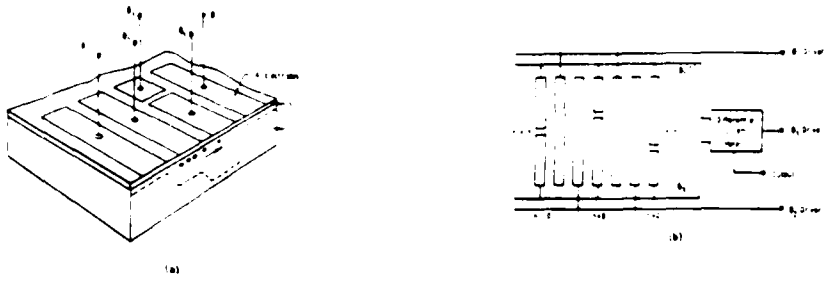


Figure 5. Schematic of the Sampling, Weighting, and Summing Operation

interruptible signal processors and as such are more compatible with the executive control required for signal processing. A 64 point CCD filter with discrete cosine transform sine and cosine chirps is shown in Figure 6. This chip was developed by Texas Instruments for the Naval Undersea Center for image processing. In addition to the four DCT filters, four DFT filters, a Hilbert transform and other experimental signal processing functions were also implemented.

Current research in CCDs is directed toward improving the charge transfer efficiency and removing the requirement of continuous "fat zero" charge injection by ion implantation techniques which keep the minority carriers away from the semiconductor oxide boundary. Ion implantation is also being used to provide asymmetric potential wells so that simpler two-phase clocking can be employed. Currently available CCDs have 500 stages with 0.9999 transfer efficiency and devices with up to 2000 stages are planned.

Another charge transfer device similar to the CCD is the Bucket Brigade Device (BBD). This is a sequence of MOS transistors coupled together by diffusion enhanced Miller capacitance. Although these devices do not operate at frequencies as high as CCDs, they have better low frequency performance since they include active devices. A CZT has been implemented with two BBD chips. Two 200 tap filters are implemented on each chip: one a discrete cosine and the other a discrete sine filter. The BBD chip is shown in Figure 7. The complex chirp used in the premultiplier and a typical input and output are shown in Figure 8. The input is an offset cosine wave and the output shows a D.C. component plus a response at the cosine wave frequency. These filters can operate at 100 kHz and have tap accuracies better than 1%. With careful control of geometry, both BBD and CCD filters with tap accuracies approaching 0.1% should be possible. This chip was also developed by Texas Instruments for the Naval Undersea Center.

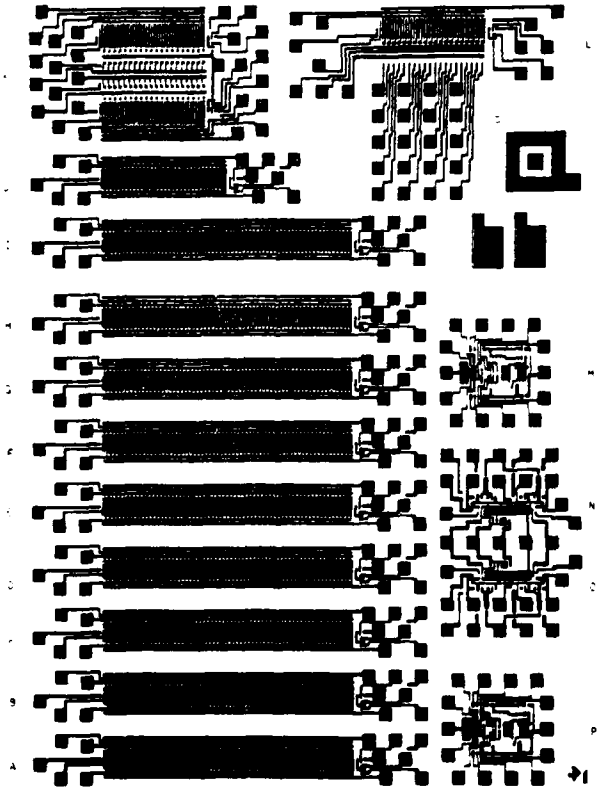


Figure 6. 64 Point CCD Filters

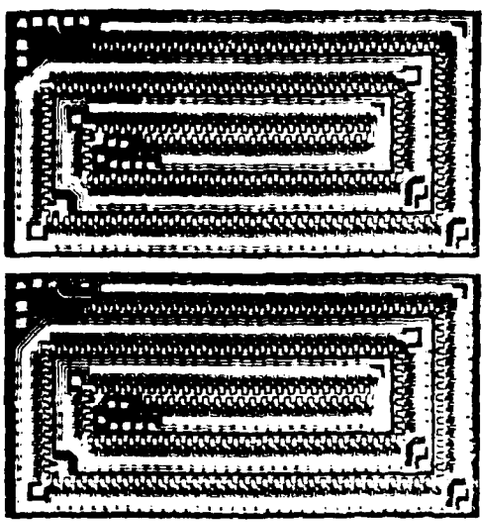
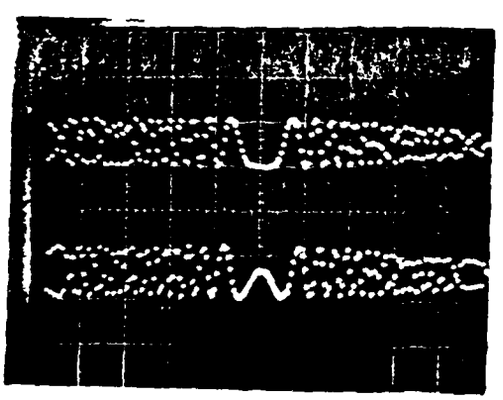
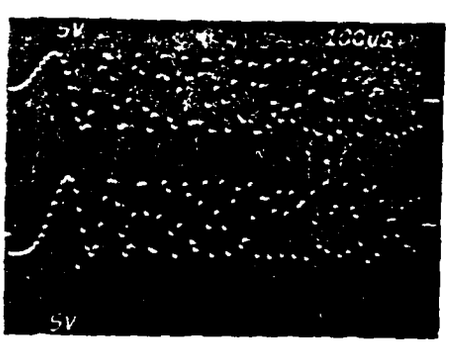


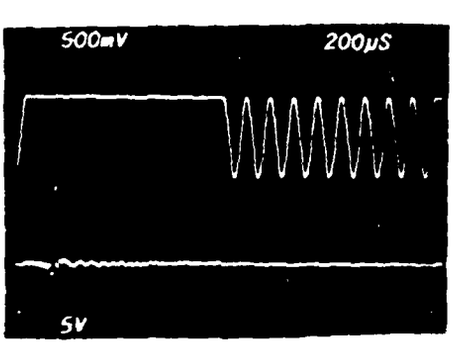
Figure 7. Bucket Brigade Chirp Filters



IMPULSE RESPONSE



PREMULTIPLIER CHIRP



INPUT TO DCT

DCT OUTPUT

Figure 8. BBD Performance

Discrete Cosine Transform

Let the data sequence be $g_0, g_1, \ldots, g_{N-1}$. The ODCT of g is defined in (5) as

$$G_k = \sum_{n=-(N-1)}^{N-1} g_n \, e^{\frac{-j2\pi nk}{2N-1}} \quad \text{for } k = 0, 1, \ldots, N-1 \tag{5}$$

The identity (15) may be used to obtain the CZT form of the ODCT shown in equation (16).

$$2nk = n^2 + k^2 - (n-k)^2 \tag{15}$$

$$G_k = 2\,R_e \left\{ e^{\frac{-j\pi k^2}{2N-1}} \sum_{n=0}^{N-1} e^{\frac{-j\pi n^2}{2N-1}} \, g_n \, e^{\frac{j\pi (n-k)^2}{2N-1}} \right\} \tag{16}$$

The block diagram of the ODCT is shown in Figure 9. Since only real inputs and outputs are required, a simplified implementation is possible and is shown in Figure 10.

A corresponding implementation may be found for the EDCT. The EDCT of g is defined by equation (9), where the extended sequence is defined by equation (10).

$$G_k = e^{\frac{-j\pi k}{2N}} \sum_{n=-N}^{N-1} g_n \, e^{\frac{-j2\pi nk}{2N}} \quad \text{for } k = 0, 1, \ldots, N-1 \tag{9}$$

$$g_{-1-n} = g_n \quad \text{for } n = 0, 1, \ldots, N-1 \tag{10}$$

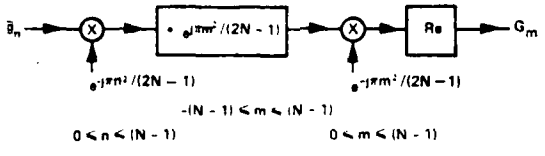


Figure 9. ODCT Block Diagram

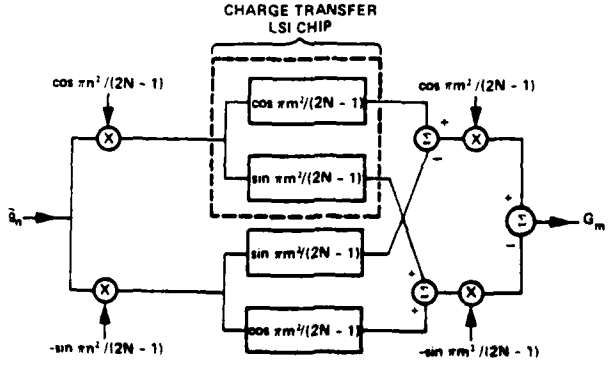


Figure 10. ODCT Expanded Block Diagram

If the mutually complex terms above are combined and the identity (15) used the CZT form of the EDCT becomes

$$G_k = 2\,\mathrm{Re}\left\{ e^{\frac{-j\pi k}{2N}} e^{\frac{-j\pi k^2}{2N}} \sum_{n=0}^{N-1} g_n\, e^{\frac{-j\pi n^2}{2N}} e^{\frac{j\pi(n-k)^2}{2N}} \right\} \qquad (17)$$

Simultaneous Computation of the DFT and the DCT

The close relationship between the DFT and the DCT permits the use of common modules to simultaneously compute both transforms. This may be accomplished most simply when an EDCT is computed using DFT modules. The sum in the EDCT defining equation (17) may be interpreted as a length 2N DFT of the extension of the function g by N zeros. This leads to the configuration shown in Figure 11. Alternatively, if the odd and even frequencies in the zero-filled DFT are considered separately, they may be computed using length N DFT modules as shown in Figure 12.

System Descriptions

Two hybrid DCT/DPCM bandwidth reduction systems have been selected for construction and evaluation. A block diagram of the systems is shown in Figure 13. The first uses a slow scan image sensor and a Bucket Brigade Device (BBD) transform implementation. The second uses an ordinary vidicon sensor and a Charged Coupled Device (CCD) transform implementation.

In the BBD system a 100 x 100 pixel solid state sensor is used. The nominal horizontal line scan time is one millisecond. The nominal frame rate is 10 frames per second which can be displayed without flicker through the use of a scan converter. The 1 millisecond line scan time was chosen in order to match the sensor to the BBD filter which operates at a clock rate of 100 kHz with good charge transfer efficiency. At 10 frames per second, image motion should be reproduced well enough for many applications, even though some picture detail is lost because of the low spatial sampling afforded by the 100 x 100 pixel format. Minimum overall bit rate is achieved by a combination of zonal filtering and variable bit assignment with low spatial frequencies assigned more bits of quantization than high spatial frequencies.(1)

The second bandwidth reduction system is compatible with a standard vidicon camera. It uses CCD filters for the cosine transform which will operate at 4.8 MHz sampling rate with a block size of 32 pixels. Compatibility with standard television format is maintained

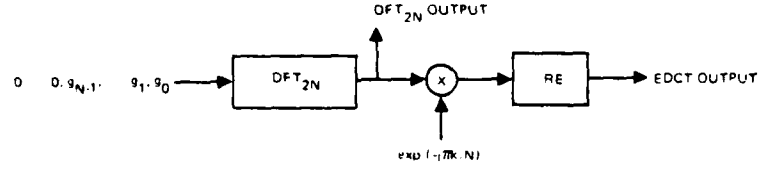


Figure 11. Computation of the DFT and EDCT Using a Single Length 2N DFT Module

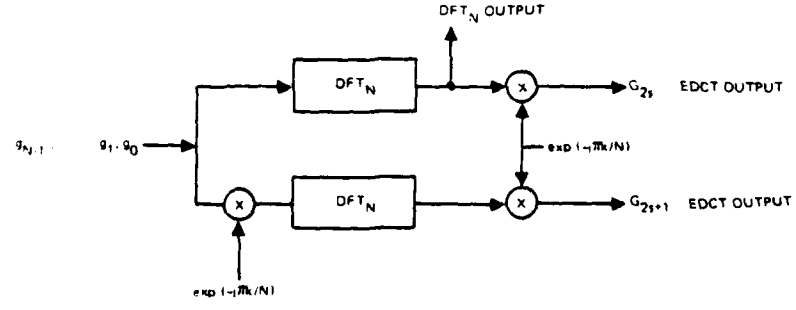


Figure 12. Computation of the DFT and EDCT Using Two Length N DFT Modules

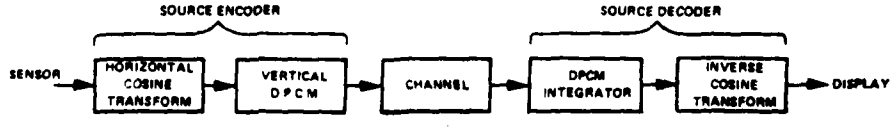


Figure 13. Image Transmission System

in as many aspects as possible. If the interlace field is used directly as the input to the transform hardware, a resolution of approximately 240 lines by 256 pixels is possible at 60 fields/sec. This is equivalent to a video bandwidth slightly less than 2.5 MHz. Computer simulation of this system is shown in Figure 14 for 1 bit pixel. P is the channel bit error rate.

The implementation of selected transforms via transversal filters was proposed at the All Applications Digital Computer Conference.[13] The computation of the discrete Fourier transform has been demonstrated using surface wave devices.[15] The discrete cosine transform has been demonstrated using bucket brigade devices.[16]

The DCT implementation block diagram is shown in Figure 10. The convolutions are performed in both television systems by charge transfer devices built by Texas Instruments. The multiplications are performed by conventional circuitry. The reference functions used in the pre- and post-multiply can be stored in a real only memory.

Figure 15 is the implementation block diagram for the DPCM part of the system. The memory is a line store which is used as the predictive element in the DPCM. The cosine transform coefficients stored in the memory are subtracted from the new transform coefficients as they enter the DPCM. The difference is quantized and transmitted with a selectable number of bits per coefficient, the number being a function of the assumed variance of the coefficient. The difference coefficients are then added to the previous line coefficients to create the predictive element for the next line.

The BBD slow scan system is shown in Figure 16. It consists of a discrete cosine transform, a DPCM, a channel simulator in which bit errors can be injected, an inverse DPCM, and an inverse DCT. The system is built on eight wire wrapped boards. A blow-up of one is shown in Figure 17. The two chips in the center are the BBD devices. System performance is illustrated in Figure 18 at 2 bits/pixel. Because the original picture has only 100 x 100 pixels, adjacent pixels are not as well correlated as in the 256 x 256 simulation and bandwidth compression algorithms do not work as well. The system performance at one bit per pixel is shown in Figure 19. Results from the CCD high resolution system are not yet available.

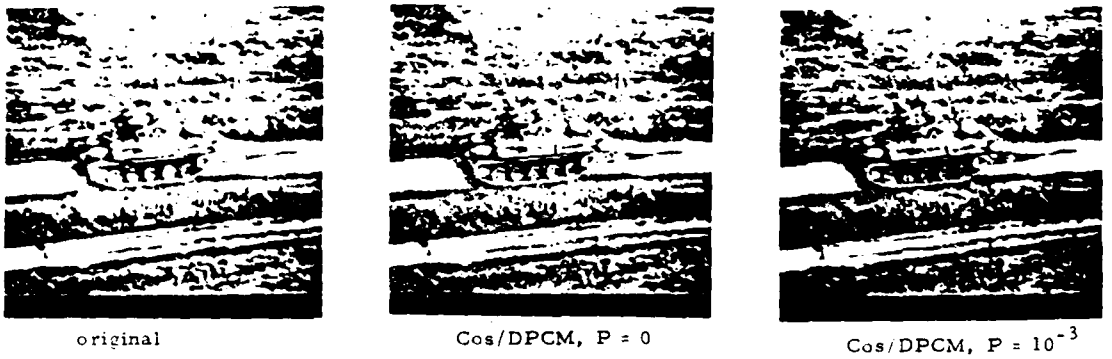


Figure 14. Computer Simulation of CCD System

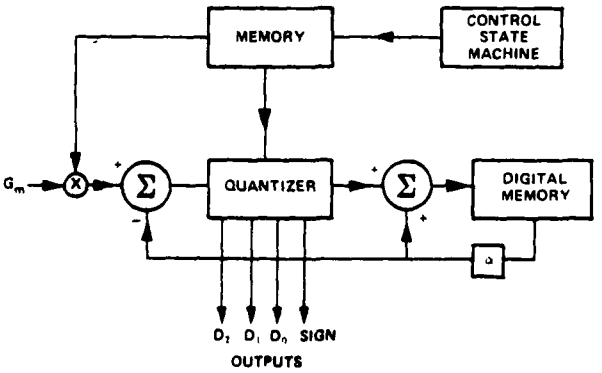


Figure 15. DPCM Encoder

Figure 16. BBD System

Figure 17 BBD Board

ORIGINAL

2 bits/pixel P = 0

2 bits/pixel P = 10^{-3}

Figure 18. BBD System Performance

1 bit/pixel P = 0

1 bit/pixel P = 10^{-3}

Figure 19. BBD System Performance

Conclusion

The use of an intraframe hybrid transformation composed of a horizontal unitary discrete cosine transform and a vertical first-order DPCM has been shown through simulation to have performance closely approximating that of a two-dimensional Karhenun-Loeve transform. This hybrid transform has been computed in real time with minimum complexity and memory through the use of LSI bucket brigade or charge coupled devices and conventional digital DPCM implementation.

Acknowledgment

This work was supported in part by Defense Advanced Research Projects Agency, Order Number 2303, Code Number 3G10, monitored by Col. H. M. Federhen.

References

1. Habibi, A. and Wintz, P. A., "Image coding by linear transformation and block quantization," IEEE Trans Commun. Technol., vol. COM-19, pp. 50-62, Feb. 1971.

2. Tasto, M. and Wintz, P. A., "A bound on the rate-distortion function and application to images," IEEE Trans. Inform. Theory, vol. IT-18, pp. 150-159, Jan. 1972.

3. Pearl, J., "Basis-Restricted Transformation and Performance Measures for Spectral Representations," IEEE Trans. on Inform. Theory, vol. IT-17, (6): pp. 751-752, Nov. 1971.

4. Davenport, W. B., and Root, W. L., Random Signals and Noise, McGraw-Hill: New York 1958, pp. 96-101.

5. Ray, W. D., and Driver, R. M., "Further decomposition of the Karhunen-Loeve series representation of a stationary random process," IEEE Trans. Inform. Theory, Vol. IT-16, pp. 663-668, Nov. 1970.

6. Pearl, J., "On coding and filtering stationary signals by discrete Fourier transforms," IEEE Trans. Inform. Theory, vol. IT-19, pp. 229-232, March 1973.

7. Ahmed, N., Natarajan, T., and Rao, K. R., "On image processing and a discrete cosine transform," IEEE Trans on Computers, C-23, pp. 90-03, Jan. 1974.

8. Habibi, A., "Hybrid Coding of Pictorial Data," Comm. Tech., COM-22, pp. 614-624, May 1974.

9. Shea, T. E., Transmission Networks and Wave Filters, D. Van Nostrand Co., 1929.

10. Bluestein, L. T., "A Linear Filtering Approach to the Computation of Discrete Fourier Transform," IEEE Trans. Audio and Electroacoustics, AU 18, pp. 451-455, 1970.

11. Rabiner, L. R., et al., "The Chirp-Z Transform Algorithm," IEEE Trans Audio and Electroacoustics, vol. AU-17, pp. 86-92, 1969.

12. Cooley, J. W. and Tukey, John W., "An Algorithm for the Machine Calculation of Complex Fourier Series," Mathematics of Computation, Vol. 19, No. 90, pp. 297-301, 1965.

13. Whitehouse, H. J., Speiser, J. M., and Means, R. W., "High Speed Serial Access Linear Transform Implementations," presented at the All Applications Digital Computer (AADC) Symposium, Orlando, Florida (January 23-25, 1973). Reprinted in Image Transmission via Spread Spectrum Techniques, ARPA Quarterly Technical Report, Order Number 2302 Code Number 3G 10 (February 1, 1973-June 1, 1973).

14. Buss, D. D., Collins, D. R., Bailey, W. H., and Reeves, C. R., "Transversal Filtering Using Charge Transver Devices," IEEE J. of Solid State Circuits, SC-8, pp. 138-146, 1973.

15. Alsup, J. M., Means, R. W., and Whitehouse, H. J., "Real Time Discrete Fourier Transforms Using Surface Acoustic Wave Devices," Proc. IEEE International Specialist Seminar on Component Performance and Systems Application of Surface Acoustic Wave Devices, Aviemore, Scotland (September 24-28, 1973).

16. Means, R. W., Buss, D. D., and Whitehouse, H. J., Real-Time Discrete Fourier Transforms Using Charge Transfer Devices, Proceedings of the CCD Applications Conference, Naval Electronics Laboratory Center, San Diego, Calif. Sept. 1973, pp. 95-101.

APPENDIX J

REAL TIME TV IMAGE REDUNDANCY REDUCTION USING TRANSFORM TECHNIQUES

REAL TIME TV IMAGE REDUNDANCY REDUCTION USING TRANSFORM TECHNIQUES*

Robert W. Means, Harper J. Whitehouse, and Jeffrey M. Speiser

Naval Undersea Center, San Diego, California 92132

ABSTRACT

Transform coding techniques have been explored extensively in theoretical studies and by simulation. It has been shown that a significant bandwidth reduction can be achieved in many applications with minimal image degradation and relative tolerance to channel errors. The major drawback of transform image coding for real time television applications in the past has been computational complexity. Transforms and algorithms have been demonstrated with recently developed charge coupled device and acoustic surface wave technologies which makes nearly optimum image transform coding feasible at real time television rates.

INTRODUCTION

Unitary transforms for image encoding have been used for intraframe encoding.[1] In addition, these techniques may also be applied to interframe and multispectral encoding. However, all unitary transformations are information preserving and no bandwidth reduction results from the application of the transform to the image. Instead, the transforms redistribute the variance associated with each picture element (pixel) so that subsequent to the transform, basis restricting operations on the transform coefficients will result in bandwidth reduction. Upon reconstruction of the original image from the basis restricted transform coefficients, a degraded version of the original image can be obtained. Unfortunately, the interrelationship between the type of transform, the form of the noninvertible operation, and the type of degradation in the reconstructed image is very complicated and subjective. The universally used analytic criterion of the mean-square-error is, at present, the best compromise technique for transform comparison.

For the particular operation of basis restriction by truncation, a particularly simple interpretation of the bandwidth reduction can be made. The transforms may be

*To be published in NATO Advanced Study Institute Series, Series E: Applied Sciences - No. 12, (Noordhoff-Leyden-1975).

viewed as a variance redistributing operation that approximately decorrelates the transform coefficients while transforming the variance associated with each picture element into the low-order coefficients of the transform. Under the assumption that each set of picture elements can be considered as a simple function from a wide sense stationary random process with correlation function $r^{|\tau|}$, there exists an optimum discrete transformation, the Karhunen-Loeve transformation, which totally decorrelates the transform coefficients and maximally compacts the variance to the low-order coefficients. All other transformations can be compared in their performance by comparing their transform coefficient decorrelation and variance compaction with this optimum transformation.

This intuitive interpretation can be made rigorous through the use of the rate-distortion criterion[2]. It has been found from experience that the closer the eigenvectors of the transformation are to the eigenvectors of the optimum Karhunen-Loeve transformation the greater the variance is compacted and the more the coefficients can be truncated while maintaining a fixed rate distortion or mean-square-error.

The use of two-dimension transforms can provide improved performance over the use of transformations on a line-by-line basis[3]. The most direct approach is to seek a two-dimensional transform which simultaneously decorrelates the transform coefficients and compacts the variance into a corner of the two-dimensional transform coefficient space. One method is to find a two-dimensional transform which can be represented as the product of a transform in one direction and a transform in the other direction. Assuming that a two-dimensional picture can be considered as a sample function from a random process with two-dimensional correlation $r_1^{|\tau_1|} r_2^{|\tau_2|}$ i.e., with a correlation coefficient r_1 in direction one and a correlation coefficient r_2 in direction two, then the optimum discrete transformation is the successive use of two Karhenun-Loeve transformations; the first with parameter r_1, and the second with parameter r_2.

Also of interest in transform encoding is block size. For a one-dimensional signal the block size is the number of elements of the transform, and the performance of the transform improves monotonically with increasing block size. For two-dimensional images, transform performance also increases with increased number of elements in each dimension of the transform. However, two dimensional transforms usually require intermediate memory to store the transform coefficients in the first direction while the transform is being computed in the second direction.

Alternatively, two-dimensional transforms can be mixed transforms, i.e., different horizontal and vertical transforms. Performance increases with the number of elements in each direction of the transform. However, for a fixed first transform size, memory requirements tend to increase linearly with the number of elements in the second transform direction since all of the coefficients must be stored from the first transform. The amount of intermediate memory is minimized by the use of a small block size for the image in the second direction, but performance depends critically on the choice of the second transform. Thus, the choice of a mixed transform interacts with the overall system design and the available memory for coefficient storage.

TRANSFORM ENCODING

Karhunen-Loeve transformation

If a continuous time function of zero mean and autocorrelation function $R(\tau) = e^{-\alpha|\tau|}$ is considered to be a sample function from a wide-sense stationary random process, then this time function can be explicitly expanded by the Karhunen-Loeve expansion[4] and the resulting coefficients will be uncorrelated. For a discrete function of zero mean and autocorrelation function $R(\tau) = r^{|\tau|}$, which may be considered as a sample function from a first-order Markov process, a similar discrete Karhunen-Loeve transformation may be defined.[5] This transformation diagonalizes the covariance matrix and is optimal in the mean-square-error sense for a restricted set of basic functions that do not span the complete space.

The discrete Karhunen-Loeve expansion is given by[5] for the case N = 2m as

$$G_k = \sum_{n=1}^{2m} \frac{2}{2m + \lambda_n^2} \sin\left\{\omega_n \left[k - (2m+1)/2\right] + n\pi/2\right\} g_n \qquad (1)$$

$$k = 1, 2, \ldots, 2m$$

where

$$\lambda_n^2 = \frac{1 - r^2}{1 - 2r\cos\omega_n + r^2} \qquad (2)$$

and ω_n are the first N positive roots of

$$\tan 2m\omega = \frac{-(1 - r^2)\sin\omega}{(\cos\omega - 2r + r^2\cos\omega)} \qquad (3)$$

Since the discrete Karhunen-Loeve expansion involves both the solution of a transcendental equation and the evaluation of the autocorrelation function of the data to be transformed, real time computation of this transform is quite complex. However, Habibi and Wintz[1] have shown that Karhunen-Loeve transformations calculated using approximate autocorrelation functions are satisfactory for many applications.

Discrete Fourier transform

Since the discrete Fourier transform is asymptotic to the Karhunen-Loeve transformation[6] for the exponential covariance function and the basis vectors are picture independent, the Fourier transform represents a logical choice for real time implementation. The Fourier transform exists for all data lengths N. It is defined by

$$G_k = \sum_{n=0}^{N-1} e^{-i\pi 2nk/N} g_n \quad k = 0, 1, \ldots N-1 \tag{4}$$

Many methods exist for the computation of discrete Fourier coefficients. The Goertzel algorithm requires a number of computations proportional to N^2 but can be used for all lengths N. When N is highly composite "fast" transformations can be used.[7] Thus, if N is of the form 2^q, then the number of computations can be made proportional to Nq. Although "fast" algorithms have been successfully used on general purpose computers, they are too slow for real time computation since the algorithm iterates q times before achieving a solution. This problem can be overcome by the use of q processors in a pipeline architecture,[8] although this increases the complexity of the processor.

A linear filter implementation also exists for the discrete Fourier transform which is both easily implemented and suitable for real-time computation. This algorithm called the chirp-Z transform[9] is based on the substitution $2nk = n^2 + k^2 - (n-k)^2$ and can be used for any length sequence N. With this substitution the DFT becomes

$$G_k = e^{-i\pi k^2/N} \sum_{n=0}^{N-1} e^{i\pi(n-k)^2/N} e^{-i\pi n^2/N} g_n \tag{5}$$

The transform may be performed as a premultiplication by a discrete chirp, convolution with a discrete chirp of twice the length, and postmultiplication by a discrete chirp. This convolution may be computed with either acoustic surface wave filters or charge transfer devices.[10]

Discrete cosine transform

Two different types of discrete cosine transform (DCT) are useful for reduced redundancy television image transmission. Both are obtained by extending the length N data block to have even symmetry, taking the discrete Fourier transform (DFT) of the extended data block, and saving N terms of the resulting DFT. Since the DFT of a real even sequence is a real even sequence, either DCT is its own inverse if a normalized DFT is used.

The "Odd DCT" (ODCT) extends the length N data block to length 2N-1, with the middle point of the extended block as a center of even symmetry. The "Even

DCT" (EDCT) extends the length N data block to length 2N, with a center of even symmetry located between the two points nearest the middle. For example, the odd length extension of the sequence A B C is C B A B C, and the even length is C B A A B C. In both cases, the symmetrization eliminates the jumps in the periodic extension of the data block which would occur if one edge of the data block had a high value and the other edge had a low value; in effect it performs a sort of smoothing operation with no loss of information. It will be noted that the terms "odd" and "even" in ODCT and EDCT refer only to the length of the extended data block in both cases the extended data block has even symmetry.

Both types of DCT may be implemented using compact, high speed, serial access hardware, in structures similar to those previously described for the Chirp-Z transform (CZT) implementation of the DFT.

Let the data sequence be $g_0, g_1, \ldots, g_{N-1}$. The ODCT of g is defined as

$$G_k = \sum_{n=-(N-1)}^{N-1} g_n \, e^{\frac{-i2\pi nk}{2N-1}} \quad \text{for } k = 0, 1, \ldots, N-1 \tag{6}$$

where

$$g_{-n} = g_n \text{ for } n = 0, 1, \ldots, N-1. \tag{7}$$

By straightforward substitution it may be shown that

$$G_k = 2 \, \mathrm{Re} \sum_{n=0}^{N-1} \tilde{g}_j \, e^{\frac{-i2\pi nk}{2N-1}} \tag{8}$$

where $\tilde{g}$ is defined by equation (8).

$$\tilde{g}_n = \begin{cases} 0.5 \; g_0, & n = 0 \\ g_n, & n = 1, \ldots, N-1 \end{cases} \tag{9}$$

The identity (10) may be used to obtain the CZT form of the ODCT shown in equation (11).

$$2nk = n^2 + k^2 - (n-k)^2 \tag{10}$$

$$G_k = 2\,\mathrm{Re}\left\{ e^{\frac{-i\pi k^2}{2N-1}} \sum_{n=0}^{N-1} e^{\frac{-i\pi n^2}{2N-1}} \tilde{g}_n \; e^{\frac{i\pi(n-k)^2}{2N-1}} \right\} \tag{11}$$

The EDCT of g is defined by equation (12), where the extended sequence is defined by equation (13).

$$G_k = e^{\frac{-i\pi k}{2N}} \sum_{n=-N}^{N-1} g_n \, e^{\frac{-i2\pi nk}{2N}} \quad \text{for } k = 0, 1, \ldots, N-1 \tag{12}$$

$$g_{-1-n} = g_n \quad \text{for } n = 0, 1, \ldots, N-1 \tag{13}$$

If the mutually complex conjugate terms in equation (12) are combined, then equation (14) results. Equation (14) may be viewed as an alternate way of defining the EDCT.

$$\begin{aligned} G_k &= 2\,\mathrm{Re}\left\{ e^{\frac{-i\pi k}{2N}} \sum_{n=0}^{N-1} g_n \, e^{\frac{-i2\pi nk}{2N}} \right\} \\ &= 2 \sum_{n=0}^{N-1} g_n \cos\left[\frac{2\pi(n+0.5)k}{2N} \right] \end{aligned} \tag{14}$$

Equation (14) may be put in the CZT format given in equation (15)

$$G_k = 2\,\mathrm{Re}\left\{ e^{\frac{-i\pi k}{2N}} \; e^{\frac{-i\pi k^2}{2N}} \sum_{n=0}^{N-1} g_n \, e^{\frac{-i\pi n^2}{2N}} \; e^{\frac{i\pi(n-k)^2}{2N}} \right\} \tag{15}$$

Ahmed[11] has investigated the use of the EDCT as a substitute for the Karhunen-Loeve transform and finds that it is superior to the Fourier transform and is comparable to the Karhunen-Loeve (K-L) in rate-distortion performance while maintaining the computation simplicity of a transform which does not depend on the picture statistics. Habibi[12] has shown by simulation that the DCT is equivalent in a mean-square-error sense to the K-L transform under basis restriction.

Original

2 Dim. Cosine 1 bit pixel P = 0

2 Dim. Cosine 1 bit pixel, P = 10^{-2}

2 Dim. Cosine 0.5 bit pixel, P = 0

2 Dim. Cosine 0.5 bits/pixel, P = 10^{-2}

Figure 1. Coded Pictures

Figure 1 shows the performance of the two-dimensional DCT on a 256x256 sampled picture with 8 bit quantization. P is the bit error probability of the transmission channel.

Hybrid transforms

Mixed transforms can be used either for intraframe encoding or interframe encoding depending on the available memory and the type of transforms implemented. In order for a mixed transform to be competitive with one of the conventional two-dimensional transforms, it must offer either superior performance or simplicity of implementation. Of the transforms examined the odd length cosine transform is competitive in performance since it can be implemented as the real part of a CZT and since the transform samples are real and are the samples of an autocorrelation function which may then be extrapolated by well-known techniques. In simulations of transform performance, the cosine transform has been shown to closely approximate the behavior of the Karhunen-Loeve transform.

The benefits of mixed transformation implementation and minimum memory may be achieved for digital transmission by combining a one- or two-dimensional unitary transform with generalized DPCM in a hybrid system. The basis of operation of the hybrid transform is that the unitary transform decorrelates the image within its constraints of transform type, dimensionality, and block size, while the generalized DPCM removes the correlation between transform blocks. This hybrid system is particularly attractive for remote sensor application since it has been found that its performance is approximately as good as the Karhunen-Loeve transform and its implementation requires minimum memory.[12]

Figure 2 shows the performance of a hybrid DCT/DPCM simulation. For a bit error probability of P = 0, there is no significant difference between the two-dimensional DCT and the DCT/DPCM. When bit errors do exist in the transmission channel, however, the hybrid system is capable of better performance. This is shown for a bit error probability $P = 10^{-2}$. Figures 3, 4 and 5 show the simulation of the hybrid DCT/DPCM performance for pictures of other scenes with different statistics.

SYSTEM IMPLEMENTATION

Two hybrid DCT/DPCM bandwidth reduction systems have been selected for construction and evaluation. A block diagram of the systems is shown in figure 6. The first uses a Charge Injection Device (CID)[13] image sensor and a Bucket Brigade Device (BBD) transform implementation. The second uses an ordinary vidicon sensor and a Charge Coupled Device (CCD) transform implementation.

In the CID system a 100x100 pixel solid state sensor will be used. The nominal horizontal line scan will be one millisecond. The nominal frame rate will be 10 frames per second which can be displayed without flicker through the use of a scan converter. The 1 millisecond line scan time was chosen in order to match the sensor to the BBD filter which operates at a clock rate of 100 kHz with good charge transfer efficiency. At 10 frames per second, image motion should be reproduced well

Cos DPCM, 1 bit pixel, P = 0

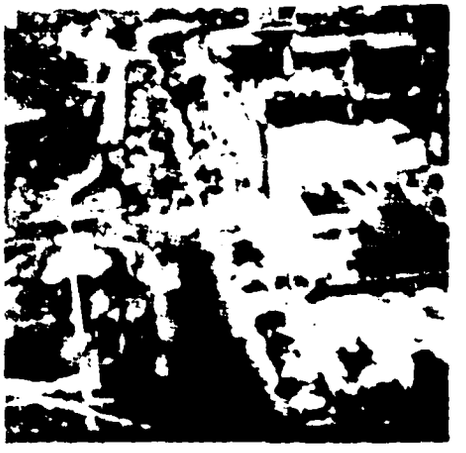
Cos DPCM, 1 bit pixel, $P = 10^{-2}$

Cos DPCM, 0.5 bits pixel, P = 0

Cos DPCM, 0.5 bits pixel, $P = 10^{-2}$

Figure 2. Coded Pictures

original

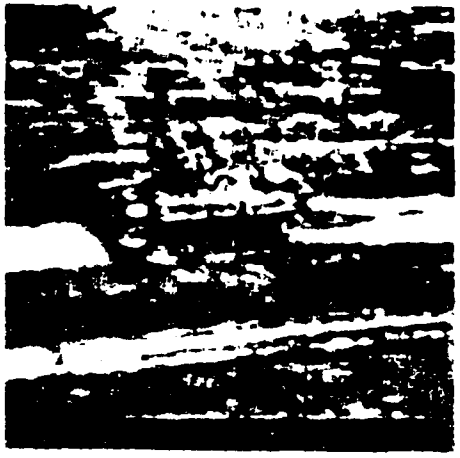


original

Figure 3. Coded Pictures

Cos/DPCM 1 bit/pixel, P = 0

Cos/DPCM 1 bit/pixel, P = 10^{-3}



Cos/DPCM 1 bit/pixel, P = 10^{-2}

Cos/DPCM 0.5 bits/pixel, P = 0

Cos/DPCM 0.5 bits/pixel, P = 10^{-3}

Cos/DPCM, 0.5 bits/pixel, P = 10^{-2}

Figure 4.

Cos/DPCM 1 bit/pixel, P = 0
Cos/DPCM 0.5 bits/pixel, P = 0
Cos/DPCM 1 bit/pixel, P = 10^{-3}
Cos/DPCM 0.5 bits/pixel, P = 10^{-3}
Cos/DPCM 1 bit/pixel, P = 10^{-2}
Cos/DPCM 0.5 bits/pixel, P = 10^{-2}

Figure 5.

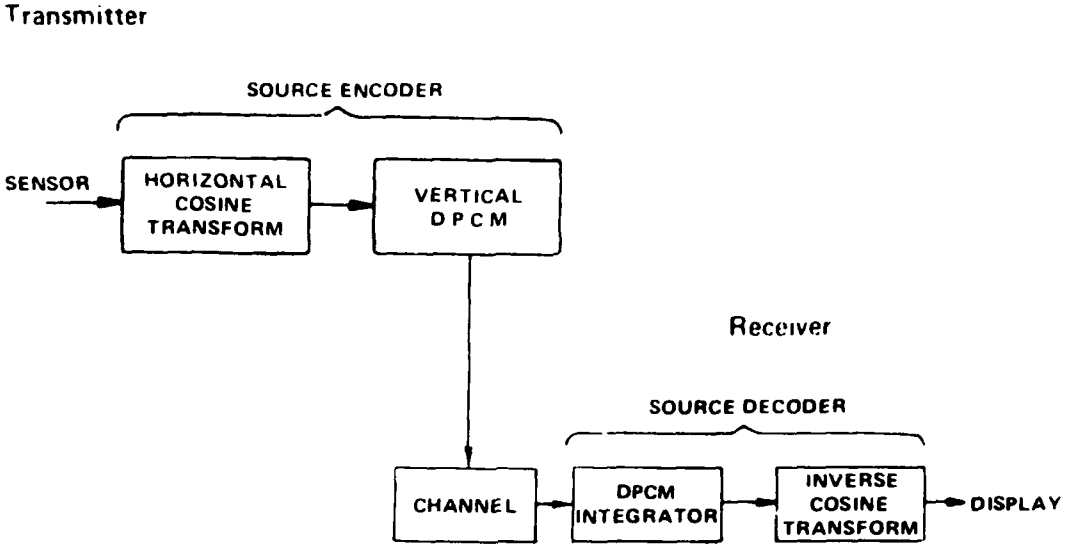


Figure 6. Image Transmission System.

enough for many applications, even though some picture detail will be lost because of the low spatial sampling afforded by the 100x100 pixel format.

Minimum overall bit rate will be achieved by a combination of zonal filtering and variable bit assignment with low spatial frequencies assigned more bits of quantization than high spatial frequencies.[1] Table 1 shows the bit rate which results from three overall bits/pixel assignments at a pixel rate of 10^5 pixels/sec. An overall bit assignment of 1 bit/pixel should result in a signal-to-distortion ratio of approximately 30 dB. In addition, since channel errors occur in the Fourier domain, channel error rates as large as $P = 10^{-2}$ will still provide useful reconstructed images.

The second bandwidth reduction system will be compatible with a standard vidicon camera. It will use CCD filters for the cosine transform which will operate at a 4.8 MHz sampling rate with a block size of 32 pixels. Compatibility with standard television format will be maintained in as many aspects as possible. If the interlace field is used directly as the input to the transform hardware, a resolution of approximately 480 lines by 256 pixels is possible at 60 fields/sec. This is equivalent to a video bandwidth slightly less than 2.5 MHz. Table 2 shows the bit rate which results for the video portion of the field.

Table 1. Bit Rate as a Function of Quantization for the CID System

Bits/Pixel	Bit Rate kilobits/sec
2	200
1	100
1/2	50

Table 2. Bit Rate as a Function of Quantization for the Vidicom Compatible System

Bits/Pixel	Bit Rate Megabits/sec
2	9.6
1	4.8
1/2	2.4

The implementation of selected transforms via transversal filters was proposed at the All Applications Digital Computer Conference.[10] The computation of the discrete Fourier transform has been demonstrated using surface wave devices.[14] The discrete cosine transform has been demonstrated using bucket brigade devices.[15]

The DCT implementation block diagram is shown in figure 7. The convolutions are performed in both television systems by charge transfer devices built by Texas Instruments. The multiplications are performed by conventional circuitry. Figure 8 is an expanded version of figure 7 showing individual components. The multipliers and adders can be put on the same chip as the transversal filters so that the complete cosine transform can be computed using two chips. The reference functions used in the pre- and post-multiply can be stored in a read only memory.

Figure 9 is the implementation block diagram for the DPCM part of the system. The memory is a line store which is used as the predictive element in the DPCM. The cosine transform coefficients stored in the memory are subtracted from the new transform coefficients as they enter the DPCM. The difference is quantized and transmitted with a selectable number of bits per coefficient, the number being a function of the assumed variance of the coefficient. The difference coefficients are then added to the previous line coefficients to create the predictive element for the next line.

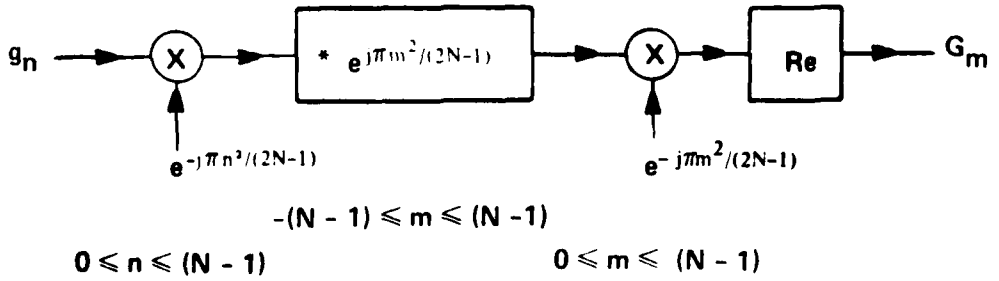


Figure 7. Serial Access Implementation of the DCT.

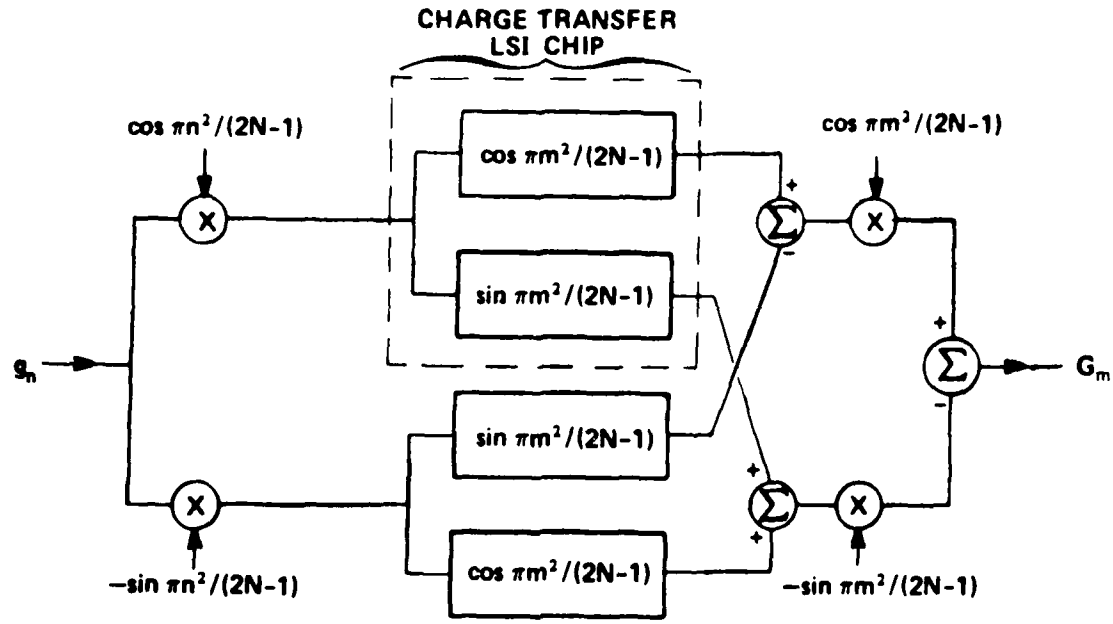


Figure 8. Charge Transfer Implementation of DCT.

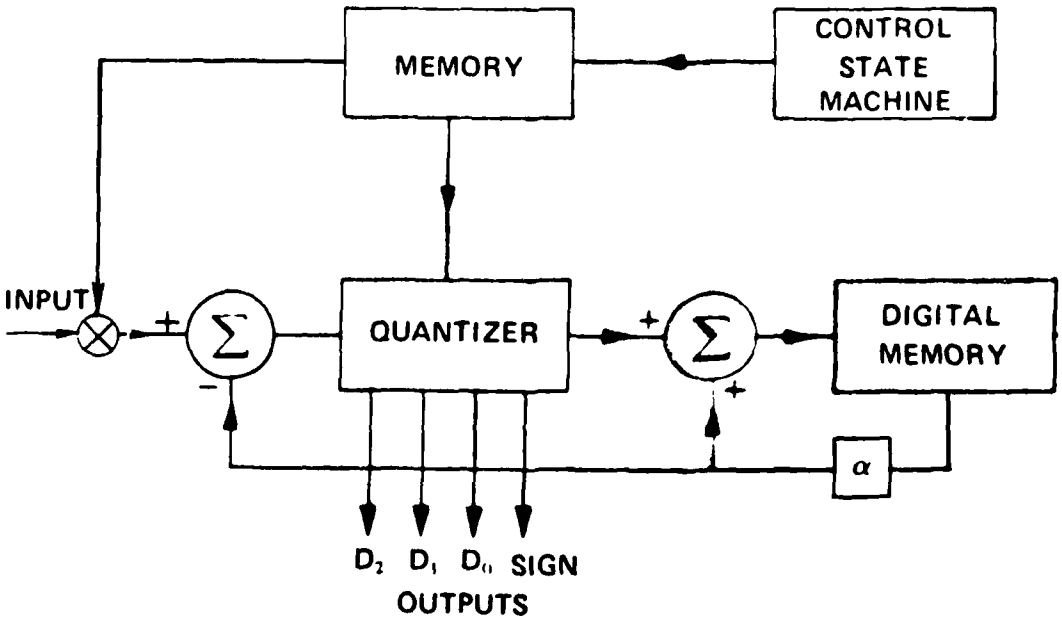


Figure 9. DPCM Encoder

CONCLUSION

The use of an intraframe hybrid transformation composed of a horizontal unitary discrete cosine transform and a vertical first-order DPCM has been shown through simulation to have performance closely approximating that of a two-dimensional Karhenun-Loève transform. This hybrid transform may be computed in real time with minimum complexity and memory through the use of LSI bucket brigade or charge coupled devices and conventional digital DPCM implementation.

REFERENCES

1. Habibi, A. and Wintz, P.A., "Image coding by linear transformation and block quantization," *IEEE Trans Commun. Technol.*, vol. COM-19, pp. 50-62, Feb. 1971.

2. Tasto, M. and Wintz, P.A., "A bound on the rate-distortion function and application to images," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 150-159, Jan. 1972.

3. Sakrison, D.J., and Algazi, V.R., "Comparison of line-by-line and two-dimensional encoding of random images," *IEEE Trans. Inform. Theory*, vol. IT-17, pp. 386-398, July 1971.

4. Davenport, W.B., and Root, W.L., *Random Signals and Noise*, McGraw-Hill: New York 1958, pp. 96-101.

5. Ray, W.D., and Driver, R.M., "Further decomposition of the Karhunen-Loève series representation of a stationary random process," *IEEE Trans. Inform. Theory*, vol. IT-16, pp. 663-668, Nov. 1970.

6. Pearl, J., "On coding and filtering stationary signals by discrete Fourier transforms," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 229-232, March 1973.

7. Cochran, W.T., et al., "What is the Fast Fourier Transform," *Proc. IEEE*, vol. 55, pp. 1664-1674, Oct. 1967.

8. Groginsky, H.L., and Works, G.A., "A pipeline Fast Fourier Transform," *IEEE Trans. Computers*, vol. C-19, pp. 1015-1019, Nov. 1970.

9. Rabiner, L.R., et al., "The Chirp-Z Transform algorithm," *IEEE Trans. Audio and Electroacoustics*, vol. AU-17, pp. 86-92, June 1969.

10. Whitehouse, H.J., Speiser, J.M., and Means, R.W., "High speed serial access linear transform implementations," presented at the 1973 Symp. All Applications Digital Computer (AADC), Orlando, Florida.

11. Ahmed, N., Natarajan, T., and Rao, K.R., "On image processing and a discrete cosine transform," *IEEE Trans on Computers*, Vol. C-23, pp. 90-93, Jan. 1974.

12. Habibi, A., "Hybrid Coding of Pictorial Data," *Comm. Tech.*, vol. COM-22, pp. 614-624, May 1974.

13. Michon, C.J. and Burke, H. N., "Operational characteristics of CID imager," *ISSCC Digest of Technical Papers*, pp. 26-27, Feb. 1974.

14. Alsup, J.M., Means, R.W., and Whitehouse, H.J., *Real-Time Discrete Fourier Transforms Using Surface Acoustic Wave Devices*, Proceedings of the IEEE International Specialist Seminar on Component Performance and System Applications of Surface Acoustic Wave Devices, held at Aviemore, Scotland, 24-28 Sept. 1973.

15. Means, R.W., Buss, D.D., and Whitehouse, H.J., *Real-Time Discrete Fourier Transforms Using Charge Transfer Devices*, Proceedings of the CCD Applications Conference held at the Naval Electronics Laboratory Center, San Diego, Calif. 18-20 Sept. 1973, pp. 95-101.

APPENDIX K

FREQUENCY SYNTHESIS V A THE DISCRETE CHIRP AND PRIME SEQUENCE ROMS

FREQUENCY SYNTHESIS VIA THE DISCRETE CHIRP AND PRIME SEQUENCE ROMS

James M. Alsup
Harper J. Whitehouse
Naval Undersea Center
San Diego, CA 92132

ABSTRACT: Surface acoustic wave sampled data filters can be utilized as serial access read-only-memories to directly implement at carrier frequencies a coherent fast-frequency-hop synthesizer in the VHF and UHF ranges. An example of frequency synthesis using surface acoustic wave discrete chirp filters is shown. A frequency synthesis scheme using a surface acoustic wave prime-sequence filter is described.

To be published in the Proceedings of the IEEE, Special Issue on Surface Acoustic Wave Devices, May 1976

FREQUENCY SYNTHESIS VIA THE DISCRETE CHIRP AND PRIME SEQUENCE ROMS

Introduction.

Many types of frequency synthesizers have been developed over the years [1], but only recently have read-only-memory (ROM) synthesizers begun to reach maturity. This has occurred primarily because of the advent of sine-cosine lookup tables implemented in digital hardware. However, other types of ROM devices are becoming available, and in particular surface acoustic wave (SAW) devices are starting to serve this purpose. ROM synthesis is particularly suited for applications which require coherent fast frequency hopping. SAW ROM technology makes possible the direct extension of ROM synthesis into the VHF and UHF ranges.

ROM Synthesis.

There are several ways to synthesisze a sampled sinusoid using discrete ROMs. One method utilizes one period of a sampled sinusoid stored in sequential order in a random access ROM [2,3]. If all samples of the ROM are read in sequence with a sample time interval Δt, then a sinusoid of period T and frequency f_0 is generated, where $T = N\,\Delta t = 1/f_0$. However, if every kth sample is read with the same sample interval Δt, then the frequency of the generated sinusoid is kf_0. Since k can take on the values 0,1,...,N-1, a total of N frequencies which are all harmonically related can be generated. The sampling ambiguity called aliasing may be evident depending upon whether complex or real sinusoids are generated.

A second method for the generation of sampled sinusoids has special application in SAW technology because it utilizes serial access ROMs. This method consists in multiplying the outputs of two N-sample discrete chirp ROMs whose serial readouts

are mutually processed by m samples to synthesize the frequency mf_0. For complex ROMs of the form $\exp(j\pi n^2/N)$, the product takes the form

$$\exp(j\pi n^2/N)\exp(-j\pi(n-m)^2/N) = \exp(j2\pi mn/N)\exp(-j\pi m^2/N) \quad (1)$$

where n corresponds to the time index and m corresponds to the frequency index. The right-hand side of this equation can be interpreted as a sampled complex sinusoid of frequency index m multiplied by a complex phase shift which is dependent only on m. Also note that $\exp(j\pi n^2/N)$ is a periodic function with period $N\Delta t$.

For frequency hopping applications, either of these two methods requires a hopping interval equal to or greater than $T = N\Delta t$ so that at least one period of each frequency is synthesized with a common nominal bandwidth less than or equal to $1/T$. Thus, the maximum hopping rate for these types of synthesizers is equal to f_0. Synthesis may take place at a carrier frequency f_c, in which case f_0 refers to the "fundamental" frequency which would be observed if the comb-like band of harmonics to be generated at the carrier f_c were shifted to baseband.

Synthesis via read-only-memory is attractive because it makes possible the coherent generation of a harmonic group of discrete frequencies. These frequencies all will start with known initial phases, and use of this a priori information can be made to build coherent receivers capable of recognizing particular phase relationships among sequentially-produced tone bursts within the operating band of the synthesizer.

ROM synthesis is also attractive because nearly instantaneous changes can be made when hopping from one frequency to the next. In contrast, a phase lock loop synthesizer invariably requires several periods of the waveform being synthesized before a stable lock is acquired.

Discrete Chirp SAW ROMs.

Discrete chirp transversal filters have been used as elements in CZT processor systems [4,5]. Such filters may be regarded as acoustic ROMs and used to implement coherent frequency synthesis. Periodic impulsing of such a SAW discrete chirp filter will result in periodic generation of the function $\exp(j\pi n^2/N$ on a carrier, so that two such ROMs operating at carrier frequencies f_1 and f_2 can provide the necessary signals to generate a tone burst over the duration of the ROM outputs. This tone is obtained by delaying one signal with respect to the other and multiplying the two ROM outputs:

$$p(t-n\Delta t)\cos[2\pi f_1 t + \pi n^2/N]\; p(t-n\Delta t)\cos[2\pi f_2 t - \pi(n-m)^2/N]$$
$$= p^2(t-n\Delta t)\cos[2\pi(f_1+f_2)t + 2\pi mn/N - \pi m^2/N] + (f_1-f_2)\text{-term}, \quad (2)$$

where $p(t)$ is the sampling window (e.g., $p(t) = 1$, $0 < t < \Delta t$, and $p(t) = 0$ otherwise). The second SAW chirp ROM output is identical to that of the first one except that it is delayed by an amount $m\,\Delta t$, and its modulation is the complex conjugate of that of the first ROM (equivalent to a negative frequency slope). Since m can vary over the interval $0,1,\ldots,N-1$ in an arbitrary order, control over the hopping sequence is flexible.

Discrete chirp ROMs are also useful since they can be used to generate a given sinusoid within the band of operation for an arbitrary length of time. This property is important for applications where frequency hopping is required on a part-time basis only. Continuous SAW chirp filters can also be used as ROMs in a similar sum or difference frequency scheme,[1] but care must be taken to account for possible discontinuities where end-points of the chirp function are encountered.

Figure 1 illustrates an experimental setup used at NUC to perform frequency synthesis using SAW discrete chirp ROMs, where $f_1 = f_2$. The base-band output waveform is shown in Figure 2, and could be filtered additionally to make the sampled waveform continuous.

[1] Some work in this area has recently come to our attention [11,12].

Figure 3 shows a proposed configuration for precise automatic control of the hopping sequence. It makes use of SAW programmable diode multipliers such as have been described in [6], and incorporates a clocking structure on the same substrate such as has been described in [7,8].

Prime Sequence ROMs.

A third method for ROM frequency synthesis involves the use of a serial access memory in conjunction with a permuter so as to achieve the equivalent of a random access ROM. Figure 4 illustrates such a configuration for the special case when the ROM is a SAW prime-sequence filter. Such a filter has been designed and constructed at NUC to implement discrete Fourier transforms via the prime transform algorithm [9,10]. This filter has as its impulse response the function $\exp(j2\pi p/N)$, where $p = R^k \bmod N$, $k = 1,2,\ldots,N-1$, and R is a primitive root of the prime number N. Thus, this SAW filter can be used as an acoustic ROM to generate permuted values of a sampled, complex sinusoid on a carrier.

The attraction of this particular ordering of the ROM samples is apparent in the structure of the auxilliary memory used to operate the tap permuter (Figure 4). This memory can be implemented in the form of a cyclically operated sequentially-addressed binary ROM whose starting position relative to the periodic impulsing of the SAW acoustic ROM determines the output frequency. N-1 frequencies can be generated, but for hopping rates slower than $f_0' = 1/(N-1)\,\Delta t$ a complex sample value (1,0) must be inserted into the synthesized output every N-1 sample intervals.

Such a SAW ROM synthesis scheme might be a reasonable alternative to the chirp method if sample permuters became available whose performance exceeded that of the chirp multipliers in terms of output accuracy.

Conclusions.

The feasibility of using surface acoustic wave devices as principal elements in ROM frequency synthesis has been demonstrated. In particular, the discrete chirp SAW device is well suited because of its ability to function at the carrier of interest, its inherent phase predictability, its fast-hop capability, and its cyclic nature in the generation of long-duration single frequency waveforms. Such SAW devices also have the characteristic properties of relatively light weight, low power, and small size, and should be considered for use as direct synthesizers in the VHF and UHF frequency regions and for those applications where the hopping rate required may exceed the capabilities of conventional digital techniques.

Acknowledgment.

This work was supported in part by Defense Advanced Research Projects Agency, Order Number 2303, Code Number 3G10, monitored by Col. H.M. Federhen.

References.

[1] Kroupa, Venceslav F., Frequency Synthesis, Halsted Press/John Wiley, New York, 1973.

[2] Tierney, Joseph, Charles M. Rader, and Bernard Gold, "A Digital Frequency Synthesizer," IEEE Trans. Audio and Electroacoustics, Vol. AU-19, March 1971, pp. 48-57.

[3] Hosking, Rodger H., "Direct Digital Frequency Synthesis," 1973 IEEE Intercon Technical Papers, Section 34/1, New York, 1973.

[4] Alsup, J.M., R.W. Means, and H.J. Whitehouse, "Real Time Discrete Fourier Transforms Using Surface Acoustic Wave Devices," Proc. IEE International Specialist Seminar on Component Performance and Systems Application of Surface Acoustic Wave Devices, Aviemore, Scotland, September 24-28, 1973.

[5] Alsup, J.M., "Surface Acoustic Wave CZT Processors," Proceedings 1974 Ultrasonics Symposium, Milwaukee, Wis., Nov. 1974, pp. 378-381.

[6] Reeder, T.M., "Electronically Variable Chirp Signal Correlation with the Diode Correlator," IEEE International Microwave Symposium Digest, Atlanta, GA, June 1974, pp. 237-239.

[7] Lee, L.L., B.J. Hunsinger, and F.Y. Cho, "A SAW-Stabilized Pulse Generator," IEEE Trans. Sonics and Ultrasonics, Vol. SU-22, March 1975, pp. 141-142.

[8] Gilden, M., T.M. Reeder, and A.J. DeMaria, "The Mode-Locked SAW Oscillator," IEEE 1975 Ultrasonics Symposium Proceedings, Los Angeles, CA, Sept. 1975, Paper P-3.

[9] J.M. Alsup, "Prime Transform SAW Device," IEEE 1975 Ultrasonics Symposium Proceedings, Los Angeles, CA, Sept. 1975, Paper G-7.

[10] Rader, C., "Discrete Fourier Transforms When The Number of Data Samples is Prime," Proceedings IEEE, Vol. 56, 1968, pp. 1107-1108.

[11] Atzeni, C., G. Manes, and L. Masotti, "Signal Processing by Analog Chirp-Transformation Using SAW Devices," IEEE 1975 Ultrasonics Symposium Proceedings, Los Angeles, CA, Sept. 1975, Paper G-6.

[12] Grant, P.M., D.P. Morgan and J.H. Collins, "Generation and Correlation of Digitally-Controlled Coherent Frequency-Hopped Waveforms Using Surface Acoustic Wave Devices," submitted to Proceedings IEEE, Letters, Oct. 1975.

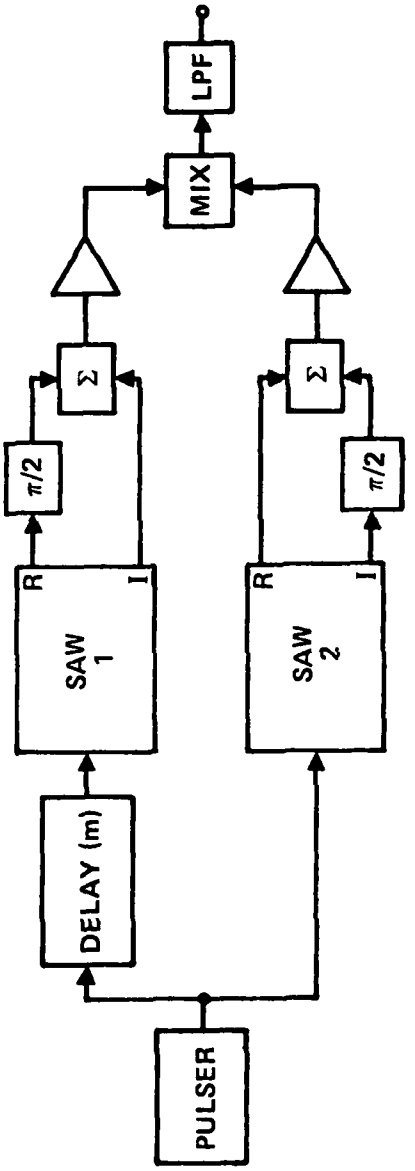


Figure 1. Frequency Synthesis Using SAW Discrete Chirp ROMs, Experimental Setup for Baseband Operation.

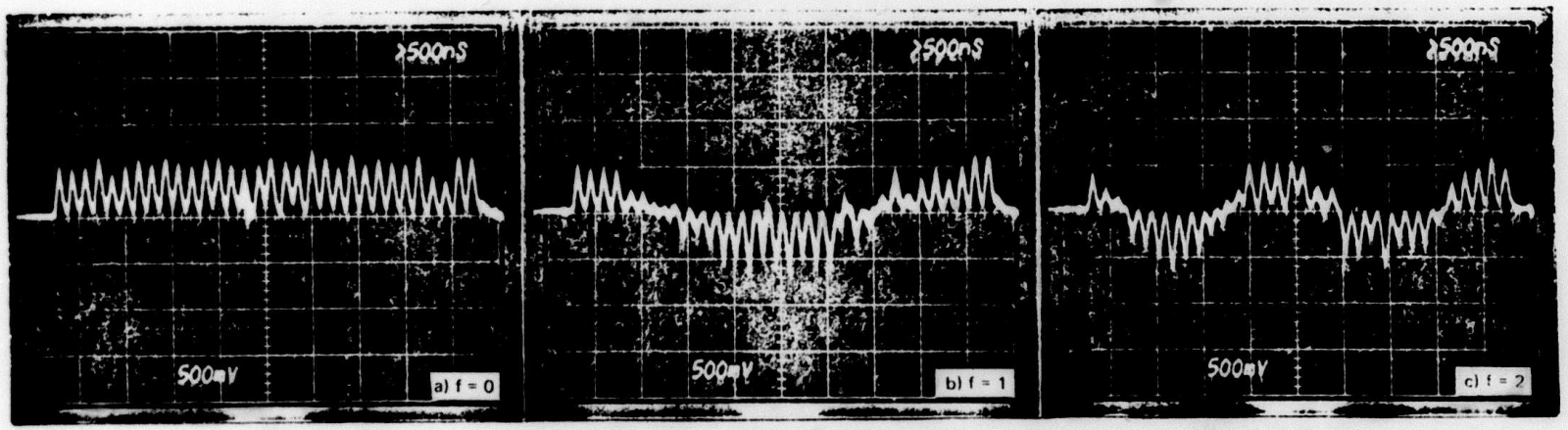


Figure 2. SAW Chirp ROM Baseband Synthesizer Outputs:
(a) m = 0; (b) m = 1; (c) m = 2.

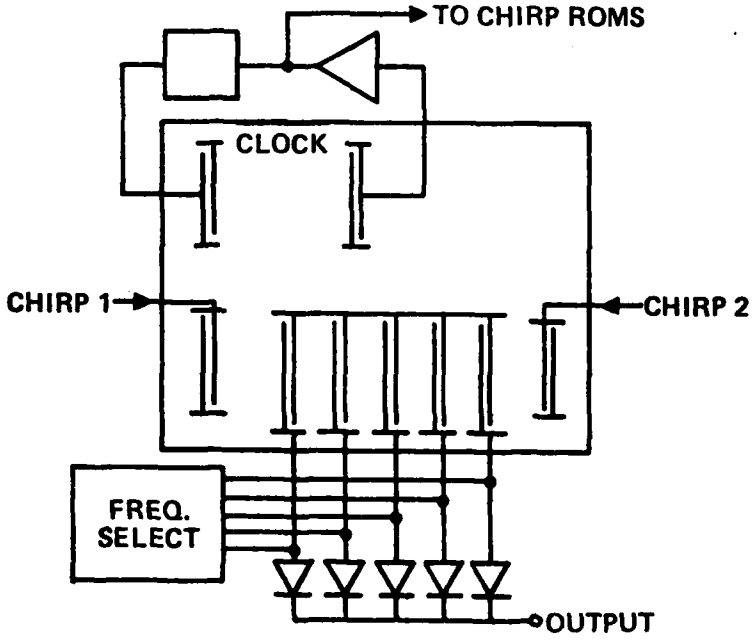


Figure 3. Automatic Frequency-Hop Controller, Proposed, For Discrete Chirp SAW Synthesizer, Uses SAW Diode-Multiplier Array with Clock and/or ROMs built onto same substrate; Output taps spaced at One-half Discrete ROM tap interval.

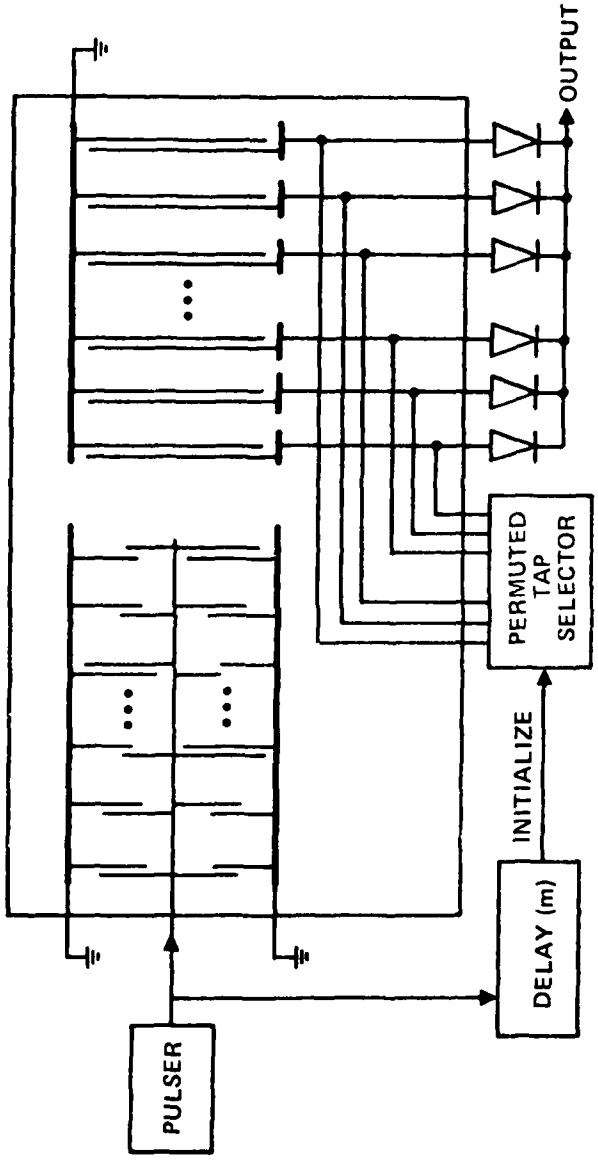


Figure 4. SAW Prime Sequence ROM Synthesizer, Proposed, using Permuter-Controlled Switchable On/off Output taps.

APPENDIX L

EXPONENTIAL RESIDUE CODES

EXPONENTIAL RESIDUE CODES

J. M. Alsup and J. M. Speiser *

Abstract - This note describes a class of pulse compression codes which were discovered by examining the properties of the prime transform algorithm [1]. For each prime P, phase hop sequences of length P-1 can be constructed whose periodic autocorrelation functions have constant sidelobe height of -1 relative to the main peak value of P-1.

For each prime P, there exists at least one integer R such that the residues modulo P of R, R^2, . . . , R^{P-1} are all distinct and non-zero, and hence form a permutation of 1, 2, . . . , P-1 [2, 3]. The number of integers possessing this property for a given prime (P) is given by the Euler ϕ-function, $\phi(P-1)$ [2, 4][1].

The code corresponding to a selected integer (primitive root) R is the sequence W^R, W^{R^2}, . . . , $W^{R^{P-1}}$, where $W = \exp(j2\pi/P)$, $j = \sqrt{-1}$. This sequence is designated an exponential residue code, since individual terms are formed by exponentiation of the residue of $(j2\pi R^n/P)$ modulo $2\pi j$, which exponent is equivalent to $(j2\pi/P)(R^n \text{ modulo } P)$. This sequence can also be recognized as a particular scrambled ordering of samples of cosine and sine functions with sampling interval inversely proportional to P.

The corresponding periodic autocorrelation function is

$$\sum_{n=1}^{P-1} W^{(-R^n)}\, W^{(R^{n+k})} = \sum_{n=1}^{P-1} W^{(R^k-1)R^n} = \begin{cases} P-1, & k \bmod (P-1) = 0 \\ -1, & k \text{ otherwise.} \end{cases} \qquad (1)$$

To prove the above identity, note that when k is between 1 and P-2 that R^k is not equal to 1, and thus (R^k-1) is non-zero. The exponents in the sum therefore run through the same P-1 vlaues (in different order) as the sequence R^n itself does, so that the sum totals -1.

To be published, IEEE Transactions on Aerospace and Electronic Systems, November 1975.

Since R^{P-1} modulo P is congruent to 1 [2], the last term in the sequence of P-1 elements is always the element W, and is always cyclically followed by the element W^R. Therefore, the codes corresponding to different primitive roots are distinct, and there are ϕ (P-1) distinct codes of length P-1.

Careful analysis of the Euler ϕ-function reveals that, for prime P>3, ϕ (P-1) is always even. Furthermore, each primitive root R has an "associate" S belonging to the set (1, 2, . . . , P-1) such that the product R·S modulo P is equal to one, i.e., $S = R^{-1}$ [2]. Thus,

$$S = R^{P-1} R^{-1} = R^{P-2} \tag{2}$$

and

$$S^n = R^{-n} = R^{P-(1+n)} \tag{3}$$

That is, the associate S is also a primitive root which can be used to generate a sequence W^S, W^{S^2}, . . . , $W^{S^{P-1}}$ which is equivalent to the sequence $W^{R^{P-2}}$, $W^{R^{P-3}}$, . . . , W^R, W. Thus, we can state that, for P>3, the ϕ(P-1) primitive roots occur in pairs such that one member of a pair generates a sequence which is the (shifted) reverse of that generated by the other member.

An exponential residue code is a shifted version of its own complex conjugate. This can be seen as follows:

$$W^{-(R^n)} = W^{R^n (P-1)} = W^{R^n R^{(P-1)/2}} = W^{R^{n'}} \tag{4}$$

where $n' = n + (P-1)/2$, and the relationships $W^P=1$; $(P-1)^2$ modulo P = 1; and R^{P-1} modulo P = 1 have been utilized. The corresponding correlation relation also holds:

$$\sum_{n=1}^{P-1} W^{(R^n)} W^{(R^{n+k})} = \begin{cases} P-1, & k \bmod (P-1) = (P-1)/2 \\ -1, & k \text{ otherwise.} \end{cases} \tag{5}$$

The autocorrelation function of the exponential residue phase hop codes may also be derived by considering Rader's prime transform algorithm [1] when the signal to be analyzed is one of the basis vectors of the discrete Fourier transform, and hence the output of the prime transform is a Kronecker delta. Other pulse compression codes are known which use the analog to a logarithm in modulo P arithmetic [5, 6]. Frequency-hop codes of a type related to exponential residue phase hop codes are also of interest and have been examined by Cooper and Yates [7].

---

[1]From [2], when a number A is written in terms of its prime factors a, b, c, etc:

$$A = a^{\alpha} b^{\beta} c^{\gamma} \text{ etc,} \tag{6}$$

then $\phi(A)$ is given by

$$\phi(A) = A \cdot \frac{a-1}{a} \cdot \frac{b-1}{b} \cdot \frac{c-1}{c} \text{ etc.} \tag{7}$$

REFERENCES

[1] Rader, C., "Discrete Fourier Transforms When the Number of Data Samples is Prime", Proc. IEEE, Vol. 56, pp. 1107-1108, 1968.

[2] Gauss, C.F. Disquisitiones Arithmeticae, (Translated by Arthur A. Clarke, Yale University Press, New Haven and London, 1966) Lipsiae, 1801.

[3] Osborn, Roger, Tables of All Primitive Roots of Odd Primes Less than 1000, University of Texas Press, Austin, 1961.

[4] Goldberg, K., M. Newman and E. Haynsworth, "Combinatorial Analysis", Chapter 24 of Handbook of Mathematical Functions, Edited by M. Abramowitz and I. Stegun, National Bureau of Standards Applied Matematics Series .55, Ninth Edition, 1970.

[5] Lerner, Robert M., "Signals Having Good Correlation Functions", 1961 Wescon Convention Record, Paper Number 9/3.

[6] Scholtz, R.A. and L.R. Welch, "Generalized Residue Sequences", International Conference on Communications, Record, Vol. I. pp. 23-1 to 23-4.

[7] Cooper, G.R. and R.D. Yates, Design of Large Signal Sets with Good Aperiodic Correlation Properties, Purdue University, School of Electrical Engineering, Electronic Systems Research Laboratory, September 1966.

* The authors are with the Naval Undersea Center, San Diego, California 92132.