

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Report 7833	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) EVALUATION OF ALGORITHMS FOR LINEAR ARRAY BEAMFORMING ON THE AN/UYK-17 (XB-1)(V) SIGNAL PROCESSING ELEMENT		5. TYPE OF REPORT & PERIOD COVERED Interim report
7. AUTHOR(s) Judith N. Froscher		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, D.C. 20375		8. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS B02-16 Department of the Navy B02-20 Naval Air Systems Director of Navy Laboratories Command, Washington, D.C. 20390 Washington, D.C. 20361		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NRL Problems B02-16; B02-20 Projects ZF11-121-003 & W21W7
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE January 28, 1975
		13. NUMBER OF PAGES 51
		15. SECURITY CLASS. (of this report) Unclassified
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Time domain beamforming                      Quantization error Frequency domain beamforming              Microprogram Dot product                                      AN/UYK-17 (XB-1)(V) Signal Processing Recursive Goertzel algorithm                Element Fast Fourier transform		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)  The purpose of this study was to determine an accurate and efficient beamforming algorithm for the AN/UYK-17 (XB-1)(V) Signal Processing Element (SPE). Briefly examining time-domain and frequency-domain linear processing shows the advantages of beamforming in the frequency domain for SPE processing. Three frequency-domain, beamforming-computation algorithms (the dot product, the recursive Goertzel, and the FFT) were examined. Because the SPE has 16-bit,  (Continued)		

two's complement, fixed-point, rounded arithmetic, a study was undertaken for two experiments: first, the case of interference by a strong coherent signal and quantization noise due to finite word length was analyzed; and second, the effect that an incoherent noise source plus quantization error has on a low strength signal was examined. The three algorithms were then assessed with regard to efficient use of the SPE's resources. Finally, as a result of these two investigations and a consideration of the tradeoffs, a beamforming macro was chosen and coded in the SPE microprogramming language.

## CONTENTS

INTRODUCTION .....	1
PASSIVE ACOUSTIC ARRAY BEAMFORMING .....	1
Beam Patterns .....	1
Time-Domain Beamforming .....	2
Frequency-Domain Beamforming .....	3
Digital Array Processing .....	3
Advantages of Frequency-Domain Beamforming for the SPE .....	4
QUANTIZATION EFFECTS IN THE SPAU BEAMFORMER ..	5
Algorithms for Frequency-Domain Beamforming .....	6
Dot Product .....	6
Recursive Algorithm .....	7
Short FFT Algorithm .....	7
The Effect of a Strong Interfering Signal .....	8
Noisy Elements .....	9
Theoretical Derivation of the Mean Power Due to Noise.	12
Theoretical Derivation of the Variance Due to Noise ..	14
Results and Comments .....	16
THE SPAU MACRO FOR FREQUENCY-DOMAIN BEAMFORMING .....	17
Pros and Cons for Each Algorithm .....	17
SPAU Macro Coded in ANIMIL .....	18
CONCLUSION .....	19
ACKNOWLEDGMENT .....	21
REFERENCES .....	21
APPENDIX A — Results for the Case of a Strong Interfering Signal .....	23
APPENDIX B — Experimental Mean and Variance For a Weak Signal Plus Noise .....	36



# EVALUATION OF ALGORITHMS FOR LINEAR ARRAY BEAMFORMING ON THE AN/UYK-17 (XB-1)(V) SIGNAL PROCESSING ELEMENT

## INTRODUCTION

The Naval Research Laboratory (NRL) is developing a microprogrammed signal processor, the AN/UYK-17 Signal Processing Element (SPE), which is intended to be part of the Navy's All Applications Digital Computer (AADC) [1]. The SPE consists of four major subsystems: a Microprogrammed Control Unit (MCU), a Buffer Store and Storage Control Unit (SCU), a Signal Processing Arithmetic Unit (SPAU), and Input/Output (I/O) unit [2]. The joint NRL/NUC Program-Controlled Processing System (PCPS) program will use the AADC SPE to perform passive acoustic array processing for a linear array [3]. The SPAU will do most of the processing for this program. The purpose of this study is to find an overall computationally efficient and accurate acoustic array processing algorithm for the SPE.

An overview of linear passive array beamforming is presented, the respective advantages of time-domain and frequency-domain beamforming are examined with processing by the SPAU in mind, and three frequency-domain beamforming algorithms are analyzed. The SPAU has two's complement, fixed-point, rounded arithmetic. Through SPAU simulation, the effects of quantization are studied for two cases. First, the effect that quantization error has on the suppression of a weak signal in the desired direction by a strong interfering signal is examined by the determination of discrepancies between "ideal" and fixed-point response curves. Second, a statistical study is undertaken to ascertain whether the error due to finite register length produces any major divergence from perfect results when a weak signal response is polluted with noise. Finally, the three frequency-domain algorithms are studied with regard to the efficient use of the SPAU's hardware.

## PASSIVE ACOUSTIC ARRAY BEAMFORMING

Because the main purpose of this study is to determine the effect of quantization on passive array beamforming, the simplest type of linear array is considered. A towed array under study for PCPS application has 34 equally spaced elements and is used for this study. A towed array is an array of hydrophones; in this case it is linear, towed underwater by a ship. Emphasis is placed on architectural considerations for the SPE to determine suitable array processing macros.

### Beam Patterns

A receiving beam is formed when an array is made to accept energy from a given direction and, to a certain degree, reject energy from all other directions [4]. The

---

Note: Manuscript submitted October 4, 1974.

directivity pattern, or beam pattern, is a graph of the output power response of the array plotted as a function of angle. Each element of an array may be weighted by coefficients, such as Taylor weights, in order to decrease sensitivity in an undesired direction; however, as a result of spatial weighting, the sharpness of the directivity pattern in the desired direction is decreased. The beam pattern for a spatially weighted symmetric array with an even number of elements  $2M$  is given by

$$\begin{aligned} K_{2M}(\theta, \phi) &= \sum_{n=-M}^M W_n e^{2\pi i(d/\lambda)(n-[1/2])(\sin \phi - \sin \theta)} \\ &= 2 \sum_{n=1}^M W_n \cos \left[ \frac{2\pi d}{\lambda} \left( n - \frac{1}{2} \right) (\sin \phi - \sin \theta) \right], \end{aligned} \quad (1)$$

where  $d$  is the distance between elements,  $\lambda$  is the wavelength of sound in water for a given frequency,  $\theta$  is the steering direction, and  $\phi$  is the source direction [5]. Equation (1) can be greatly simplified, as shown later, for uniform weighting. The number of elements, the element spacing, and the wavelength determine the beam pattern. The main lobe beamwidth and the side lobe height determine the quality of the beamformer—generally, the narrower the main lobe and the lower the side lobes, the better the beamformer. Adaptive procedures can sometimes be applied and improved results obtained. This study concentrates on an unshaded linear array. Uniform weighting was chosen because of the associated simple and well-understood array response. Emphasis is placed on architectural considerations for the SPE to determine suitable array processing macros.

### Time-Domain Beamforming

Perhaps the theory of beamforming is easier to understand for the time domain since the algorithm is a simple delay and sum procedure. Suppose that the beamformer is steered at  $\theta$  degrees. This corresponds to rotating the array through  $\theta$  degrees. Now, when the signal impinges upon the array in a direction normal to the array before rotation, the  $(m - 1)$ th element receives the signal a time  $[d \sin(-\theta)]/c$  later than the  $m$ th element. Hence, the signal at each element must be delayed by  $[-(m - 1)d \sin \theta]/c$ , where  $m$  is the element number, in order to steer the beamformer to  $\theta$  degrees. A conventional time domain beamformer with eight elements is shown in Fig. 1, in which  $\tau$  is the time delay between elements.

A beamformer steered at broadside has  $\theta = 0$ . The main lobe of the beam pattern is in the  $\theta = \sin^{-1}(c\tau/d)$  direction. The equation for the time domain beamformer is

$$R(t) = \sum_{n=0}^{2M-1} W_n s(t - n\tau), \quad (2)$$

where  $W_n = 1$  for uniform spatial weighting.

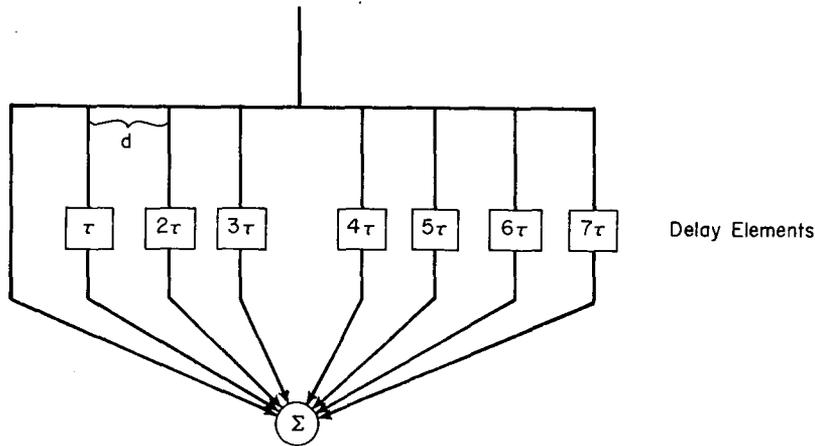


Fig. 1—A conventional time-domain beamformer

### Frequency-Domain Beamforming

By means of the Fourier transform beamforming can be performed in the frequency domain. Both types of beamforming, while they are equivalent through the Fourier transform, have their respective advantages; hence, one type can be chosen as being more desirable depending upon the requirements of the system.

Now with uniform weighting, let us Fourier transform the time domain beamformer equation:

$$\hat{R}(\omega) = \int_{-\infty}^{\infty} R(t)e^{-i\omega t}dt = \sum_{n=0}^{2M-1} \hat{S}(\omega, n)e^{(-2\pi ind \sin \theta)/c}$$

where  $\hat{S}(\omega, n)$  is defined to be the Fourier transform of the signal at the  $n$ th element and  $\omega = 2\pi f$ . Therefore, a time delay in the time domain is equivalent to multiplication by a phase factor in the frequency domain.

### Digital Array Processing

The development of special purpose high-speed digital processors has made it desirable to perform signal processing algorithms digitally. With digital implementation of signal processing procedures, automatic gain control requirements are reduced since, for example, a 90-dB range is provided by a 16-bit machine such as the SPAU. Also, digital processing is more flexible because filter coefficients, etc., can be changed simply by changing the contents of digital registers. By using a parallel architecture in the processor, relatively high frequency signals can be processed—the limit is set typically by the speed of the A/D converters [6]. For very low frequency signals, analog elements (capacitors, inductors) are very large; hence, digital techniques can be used to advantage [6]. Procedures cannot be exactly repeated by analog means because analog device values are sensitive to environmental conditions, whereas digital states are not. Components in an analog system can be faulty and go undetected because the system will usually continue

to function with some attendant error; whereas, in a digital system, if a component is faulty, the whole system usually fails to operate. Although digital techniques introduce quantization errors, they are, in general, more accurate than analog processes because precision of electrical components in analog devices is limited. The introduction of the fast Fourier transform (FFT) [7], a *fast* algorithm for calculating the discrete Fourier transform of a sequence, has also added to the impetus for processing in the digital domain.

### Advantages of Frequency-Domain Beamforming for the SPE

Usually, the output of a time-domain beamformer is transformed to the frequency domain for subsequent spectral analysis. The output of the frequency-domain beamformer is already in the frequency domain. There are, however, computational advantages for the SPE in performing frequency-domain beamforming. The SPAU contains four high-speed multipliers and associated adders and thus has the capability for parallel processing. Complex multiplications and FFTs are operations that are well equipped to use efficiently the SPAU's hardware. Frequency-domain beamforming requires the kind of operations that the SPE is designed to do. The time-domain approach is a delay and sum process with multiplications if the array is spatially weighted. Beamforming in the time domain requires the dynamic storage of signal data for the purpose of effecting time delays on these data. To obtain the required delay resolution (i.e., to steer in small angular increments), it is necessary to interpolate the sampled signal or to greatly oversample. To obtain a sufficiently small time delay for the specified resolution by interpolation is very costly and also slows processing time considerably, and to oversample requires faster analog-to-digital (A/D) conversion and larger amounts of storage, which are very expensive. It is informative to calculate the time delay required for a particular array so that it can be steered in increments of half the half-power main lobe width.

Equation (1) can be simplified by means of Eqs. (417) and (418) from Ref. 8 if uniform weights are assumed. Hence,

$$K_{2M}(\theta, \phi) = \frac{\left[ \sin 2M \left( \frac{2\pi d}{\lambda} \right) \left( \frac{\sin \phi - \sin \theta}{2} \right) \right]}{\sin \left[ \left( \frac{2\pi d}{\lambda} \right) \left( \frac{\sin \phi - \sin \theta}{2} \right) \right]}$$

where  $2M = 34$  and  $\lambda = c/f$ . The object is to solve for  $\phi$  when

$$K_{2M}(\theta, \phi) = \frac{2M}{\sqrt{2}}.$$

Let

$$x = \frac{2\pi d}{\lambda} (\sin \phi - \sin \theta).$$

$$\frac{\sin \frac{34}{2} x}{\sin \frac{x}{2}} = 17 \sqrt{2}.$$

By the method of successive approximations,

$$x = 0.08189 \quad \text{and for} \quad \theta = 0,$$

$$\sin \phi = 0.08189 \frac{\lambda}{2\pi d}.$$

Now  $\tau_\phi = d \sin \phi / c$  seconds.

To satisfy the Nyquist criteria, the signal would be sampled only every  $\tau_s = 1/2f$  seconds. Now,

$$\frac{\tau_s}{\tau_\phi} = \frac{c 2\pi d}{2fd(0.08189\lambda)} = \frac{\pi}{0.08189} \approx 38.5.$$

Thus, for the time-domain formulation, it would be necessary to sample 38.5 as many times as normal in order to steer the beam in increments of half the half-power main lobe width. This implies that for a given bound on accuracy up to 38 times as much delay storage is needed for time-domain beamforming, whereas for the frequency domain all that is required is a complex multiplication by a phase factor. Hence, for the SPE, frequency-domain array processing is more efficient in utilizing hardware, requires less storage, and is better matched to the processing capabilities of the SPAU.

## QUANTIZATION EFFECTS IN THE SPAU BEAMFORMER

Finite register length imposes a limitation on the computational accuracy in digital processing. As noted earlier, however, the error from quantization is, in general, less than the inaccuracies from processing by analog means. In this study, the effects of finite register length are examined only for the frequency-domain beamforming algorithms; quantization error for the FFT which transforms the time-domain samples to the frequency domain where they are processed has been analyzed in another report [9]. In a two's complement, fixed-point, rounded, 16-bit machine, quantization error is introduced in the following ways:

1. Quantization of input data
2. Quantization of system coefficients such as sines and cosines used in complex multiplication
3. Rounding at the multipliers
4. Scaling to prevent overflow in the adders.

The first two sources of error are obvious. When two 16-bit numbers are multiplied, a 32-bit product is obtained; hence, this product must be rounded to 16 bits so that the

processing can continue. To prevent overflow, the inputs to adders must be scaled for fixed-point machines. Scaling can cause large errors. It is usually performed before the input data goes through an algorithm; hence, the whole word length is not fully utilized. In some algorithms, such as the FFT, when scaling is needed, it is done on blocks of data at certain stages of the computation. This method kicks out a “bit” of error at various stages in the calculation and uses the word length more efficiently. The FFT algorithm is done in stages, the results of each stage are tested for overflow conditions, and the whole block of data is scaled by 1/2 if the test indicates the need for scaling.

### Algorithms for Frequency-Domain Beamforming

A frequency-domain beamformer is shown in Fig. 2.

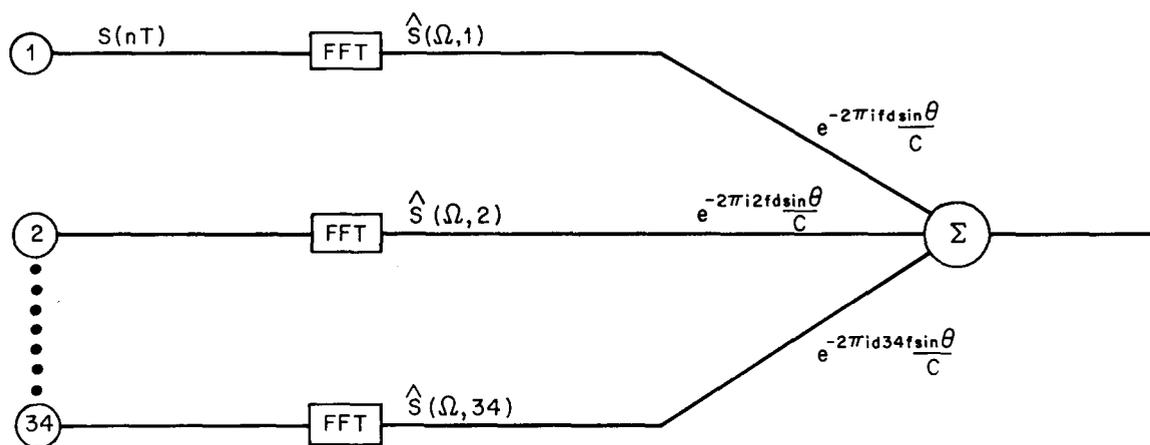


Fig. 2—Frequency-domain beamformer

A block of points from each sensor is transformed via the FFT to the frequency domain. To form a beam in the  $\theta$  direction, the Fourier transform of a signal from each sensor is chosen, each array is at the same frequency, and each output is multiplied by the appropriate complex phase factor and then summed over the elements. The equation for the frequency-domain beamformer is

$$R(\theta) = \sum_{k=1}^{34} \hat{S}(\Omega, k) e^{(-2\pi i f k d \sin \theta) / c} \quad (3)$$

where  $\theta$  is the steering angle,  $k$  is the sensor number, and  $\hat{S}(\Omega, k)$  is the frequency-domain signal from the  $k$ th sensor with frequency  $f = \Omega / 2\pi$ . We will examine three ways in which the SPAU can implement the operation defined by the right side of Eq. (3).

*Dot Product*—The dot product algorithm is perhaps the most straightforward of the three since it is the frequency-domain beamforming equation. The CDC 6400-6600 KRONOS 60-bit floating-point implementation of the beamformer equation will be considered ideal. Let  $\bar{U}$  be the complex vector

$$\bar{U} = \begin{bmatrix} e^{(-2\pi i f d \sin \theta)/c} \\ e^{(-2\pi i 2 f d \sin \theta)/c} \\ \vdots \\ e^{(-2\pi i 34 f d \sin \theta)/c} \end{bmatrix}$$

and let  $\bar{S}$  be the  $1 \times 34$  complex vector

$$\bar{S} = [\hat{S}(\Omega, 1) \hat{S}(\Omega, 2) \dots \hat{S}(\Omega, 34)]$$

where  $\Omega = 2\pi f$  and  $k$  refers to the element position. Now the response is

$$R(\theta) = \bar{S} \cdot \bar{U},$$

or

$$R(\theta) = \sum_{k=1}^{34} \hat{S}(\Omega, k) e^{(-2\pi i k f d \sin \theta)/c}.$$

If  $|\hat{S}(\Omega, k)| \leq 1 \forall k$ , then  $|R(\theta)| \leq 34$  by the triangle inequality and the Cauchy-Schwartz inequality. To prevent overflow in the fractional, fixed-point arithmetic of the SPAU, the input (i.e.,  $S(\Omega, k)$ ,  $k = 1, \dots, 34$ ) is scaled by 32 and, hence, 5 bits are lost before the computation begins. Scaling by 34 would absolutely prevent overflow; however, overflow was not encountered with the input scaled by 32. This method requires a complex multiply for each sum in the summation and, hence, uses all four SPAU multipliers.

To form a beam in the  $\theta$  direction by the dot product requires 34 complex multiplications, 68 real additions (2 for each complex multiplication), 34 complex additions and the access of 34 coefficients,  $\exp [(-2\pi i k f d \sin \theta)/c]$ .

*Recursive Algorithm*—Another algorithm that is somewhat more efficient is a recursive form of the Goertzel algorithm for calculating the discrete-Fourier transform (DFT) of a sequence, one point at a time [10]. The recursive form is

$$y(0, \Omega) = (0, 0)$$

$$y(k, \Omega) = y[(k-1), \Omega] e^{(2\pi i f d \sin \theta)/c} + \hat{S}(k, \Omega)$$

where  $k = 1, 2, \dots, 34$  and  $R(\theta) = y(34, \Omega)$ . This procedure requires 34 complex multiplications, 68 real additions (2 for each complex multiplication), 34 complex additions, but only one coefficient  $\exp [(2\pi i k f d \sin \theta)/c]$ . Again, scaling is performed on the input, which increases the error due to quantization because the full word length cannot be used.

*Short FFT Algorithm*—The third approach is to use a short FFT to calculate the sum [11]. This algorithm is very fast and accurate. Since 34 is not a power of 2, at least a 64-point FFT, zero extended, must be used. Let  $N$  be the number of points used

for the FFT. When the FFT is used to form a beam in the frequency domain,  $R[(Nfd \sin \theta)/c]$  is calculated and  $R(\theta)$  must be obtained from the FFT output. To obtain the required resolution in the beam pattern, a 256-point FFT is needed. Although the FFT is faster than the recursive algorithm or the dot product for the same number of points, it is not faster when a 256-point FFT must be computed. For  $N$  a power of 2,  $N \log_2 N$  real additions and  $(1/2)N \log_2 N$  real multiplications are needed to compute the FFT of  $N$  points [10]. Let us compare the number of real multiplications needed for the three algorithms:  $4 \times 34 = 136$  real multiplications for both the dot product and the recursive algorithm and at most  $(1/2)(256)(8) = 1024$  real multiplications for the FFT. Another disadvantage encountered when using the FFT is that the response for any arbitrary steering angle cannot be obtained. Although the FFT is not as flexible as the other algorithms with regard to steering direction, it is very accurate and can use a block scaling technique that minimizes the quantization error due to scaling. Using a 256-point FFT is not desirable, however, because it slows the computation of  $R(\theta)$  significantly. An alternative approach would be to use a 64-point FFT, which would limit the beamformer's ability to steer in certain directions—it would coarsen the resolution.

### The Effect of a Strong Interfering Signal

To examine the effects of quantization on frequency-domain beamforming, the response due to a weak signal in the look direction plus a strong interfering signal is computed in both 16-bit fixed-point and 60-bit floating-point arithmetic. After choosing four steering directions, the beam pattern for each direction is examined to determine the placement of the interfering signal. A signal of varying magnitude is placed in the steering direction and is expected to have a stronger influence on the array response than the interfering signal. An interfering signal of constant magnitude (0.5) is placed at a peak or a null of the beam pattern for each respective steering direction; thus, it will have less effect if it is in a null than if it is placed at a peak. Both signals are sinusoidal and at the same frequency.

The purpose of this experiment is to determine whether quantization causes large discrepancies between the ideal floating-point response and the SPAU simulated fixed-point response. The magnitude of the signal in the steering direction is varied from a value of 0.5 to 0. Although it is customary in beamforming analysis to keep the magnitude of the signal in the steering direction normalized to 1, it is not practical when executing in a fixed-point processor because overflow is always a problem and increasing the magnitude of the interfering signal while keeping the magnitude of the signal in the look direction constant would eventually result in overflow.

The simulated output from each sensor is generated, using floating-point arithmetic, from the amplitude and phase of the two interfering signals and is then fed into one of the beamforming algorithms. The ideal response is calculated using the dot product algorithm on the CDC 6400-6600 KRONOS system with 60-bit floating-point arithmetic. To gain confidence in the computer results, a theoretical closed-form expression for the response of an array steered at  $\theta$  degrees with the source at  $\phi$  degrees is calculated for an amplitude of 0.5.

$$R(\theta, \phi) = 2 \sum_{k=1}^{17} 0.5 \cos \left[ \frac{2\pi d}{\lambda} \left( k - \frac{1}{2} \right) (\sin \phi - \sin \theta) \right]$$

where  $\lambda = c/f$  and  $d$  is the element spacing. This equation yields the response due to one signal. To obtain the response due to two signals, it is necessary to multiply by an appropriate phase factor so as to cause both signals to seem to be from the same direction. The response due to the interfering signal alone agrees with the program output. The three beamforming algorithms are SPAU simulated in both rounded and truncated arithmetic and the sensor outputs are scaled by 32. The four steering angles are  $0^\circ$ ,  $3^\circ$ ,  $15^\circ$ , and  $45^\circ$ , and both signals are at the same frequency.

For each steering angle and each interfering source direction, the magnitude response due to the two signals is plotted versus the magnitude of the signal in the look direction. To calculate the magnitude response, the real and imaginary parts of the response are squared, summed, and then raised to the one-half power. When the magnitude of the signal in the steering direction is very small, the difference between the ideal response and the SPAU response is greater for all three algorithms, the shape of the fixed-point curve is different from the ideal, and the fixed-point curve is actually discontinuous for responses of very small magnitude. These discontinuities are caused by thresholding, or underflow. This phenomenon occurs when a number is too small to be represented in 16 bits. The output from the beamformer is in complex form. For a very small signal in the source direction and the interfering signal in a "low" of the beam pattern, the output from the beamformer is very small. Hence, when the output is squared, it is even smaller and cannot be represented in 16 bits—a great deal of significance relative to the squared number is lost due to rounding or truncation. Figure 3 dramatically exhibits the thresholding effect. As the squared value attains another quantization level, the magnitude response steps up in a discontinuous manner. The beam is steered at  $0^\circ$  and the interfering signal is at  $9^\circ$ , a low point of the beam pattern, and the results are from the recursive algorithm.

In Figs. 4 and 5, results from the recursive algorithm are plotted with the beamformer at  $15^\circ$  and the interfering signal at  $22^\circ$ . The magnitude of the response of the two signals is plotted as a function of the amplitude of the signal at  $15^\circ$ . In Fig. 4, the amplitude of the signal at  $15^\circ$  ranges from 0.0 to 0.5. With this resolution, the SPAU simulated results are congruent with the floating-point ideal responses.

In Fig. 5, the region from 0.0 to 0.01 in the abscissa is magnified and the difference in the floating-point and fixed-point curves can be seen. Other curves are shown in Appendix A, together with all the results in tabular form. The SPAU simulated results agree very well with the ideal results. As was observed earlier, exact values of the steering angle cannot be obtained with the FFT; hence, response at the angle closest to the steering angle is recorded. The recursive algorithm proved to be more computationally efficient and acceptable with regard to accuracy.

## Noisy Elements

The sensor elements themselves are subject to random acoustic and electrical noise and it is of interest to consider how this noise affects the beamformer's sensitivity in the steering direction. This study examines the degree to which the effects of finite register length coupled with a noise source whose power spectra has a constant magnitude and a random phase suppress a weak signal in the steering direction. Because the SPAU was ultimately designed with rounded arithmetic, all simulations are performed for rounded

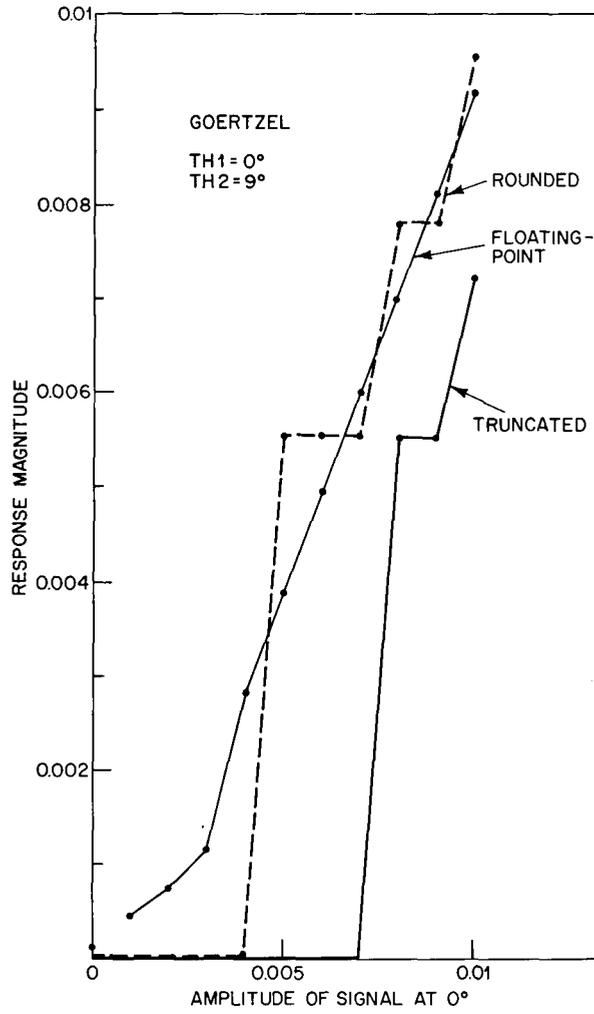


Fig. 3—Signal strength at steering direction, 0°, vs magnitude of response with interfering signal at 9°

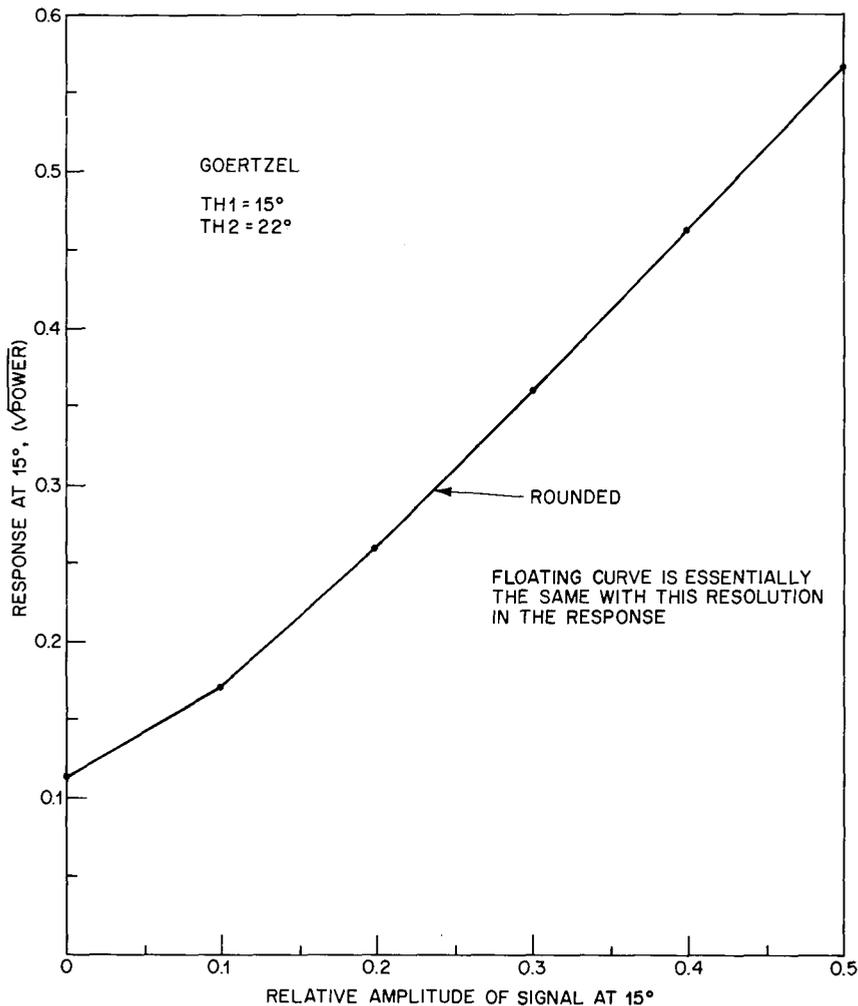


Fig. 4—Signal strength at steering direction of 15° vs magnitude of response with interfering signal at 22°

arithmetic. The KRONOS random number function was used to generate the random phases ( $0$  to  $2\pi$ ) of the noise samples. A noise sample is added to the signal sample of each hydrophone element of the array. Then it is fed into the ideal dot product beamformer, the SPAU simulated dot product beamformer, and the SPAU simulated recursive beamformer. Since it is desirable to obtain a statistically stable mean response and since many iterations (5000) are performed to obtain a stable mean, the FFT was not simulated because it is not flexible with regard to steering angle and because of the computer expense that would be incurred in such a simulation. The same four steering directions as used in the first experiment are used for this study. The magnitude of the signal in the steering direction is from 0.0 to 0.1 and both the signal and the noise are scaled by 32. The array parameters are also the same. By experimental means, the magnitude range was chosen from 0.0 to 0.1 since for larger magnitudes the effects of uncorrelated noise are similar for both 16-bit fixed-point and 60-bit floating-point arithmetics. As evidenced by the previous experiment, the noticeable discrepancies between “ideal” and fixed-point occur for small signal strengths.

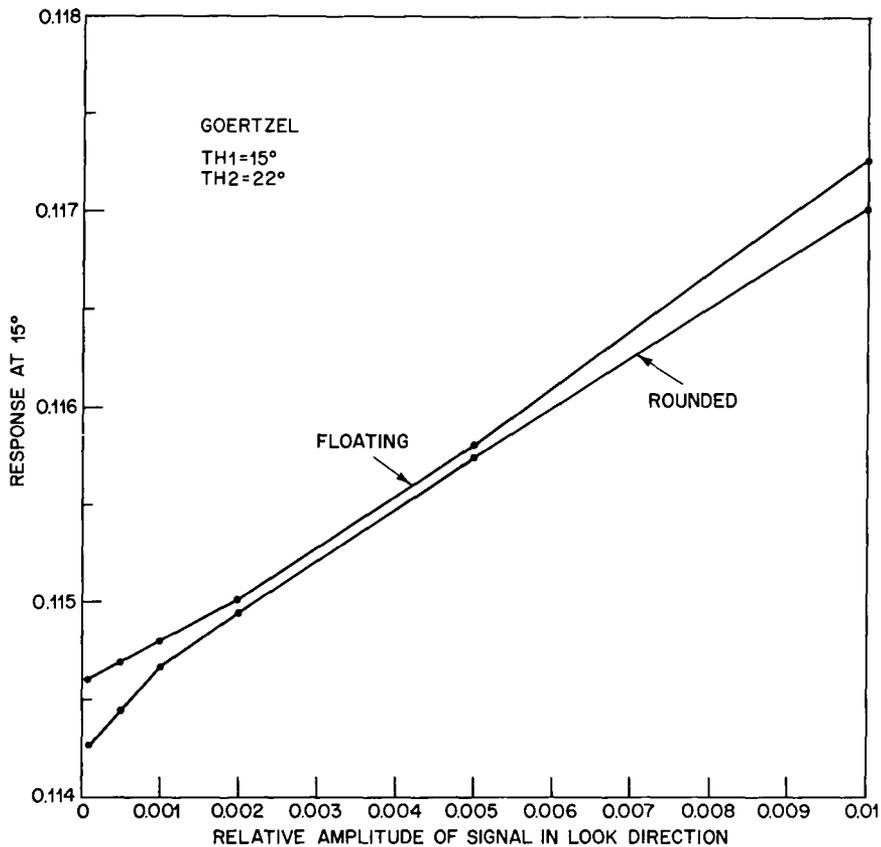


Fig. 5—Signal strength at steering direction of 15° vs magnitude of response with interfering signal at 22° (magnified)

The object of the study is to examine the effects that quantization has on the response due to signal plus noise. The following scheme (Fig. 6) is used for this experiment.

There are 5000 iterations of the beamforming algorithm used to obtain a variance of  $10^{-6}$ . Since there are 34 elements, the random number generator is called 34 times for each iteration. The randomly phased noise is added to the hydrophone signal and is then run through the three beamforming algorithms where the power response is calculated. The mean power for each signal strength in a particular direction is calculated and plotted against signal magnitude.

*Theoretical Derivation of the Mean Power Due to Noise*—We will compute the mean of the beamformer output due to noise alone. The mean power with a nonzero signal is just the sum of the power due to the signal alone and the mean power due to the noise since no random variable is involved in the calculation of the mean of the signal. The equation for the complex response is

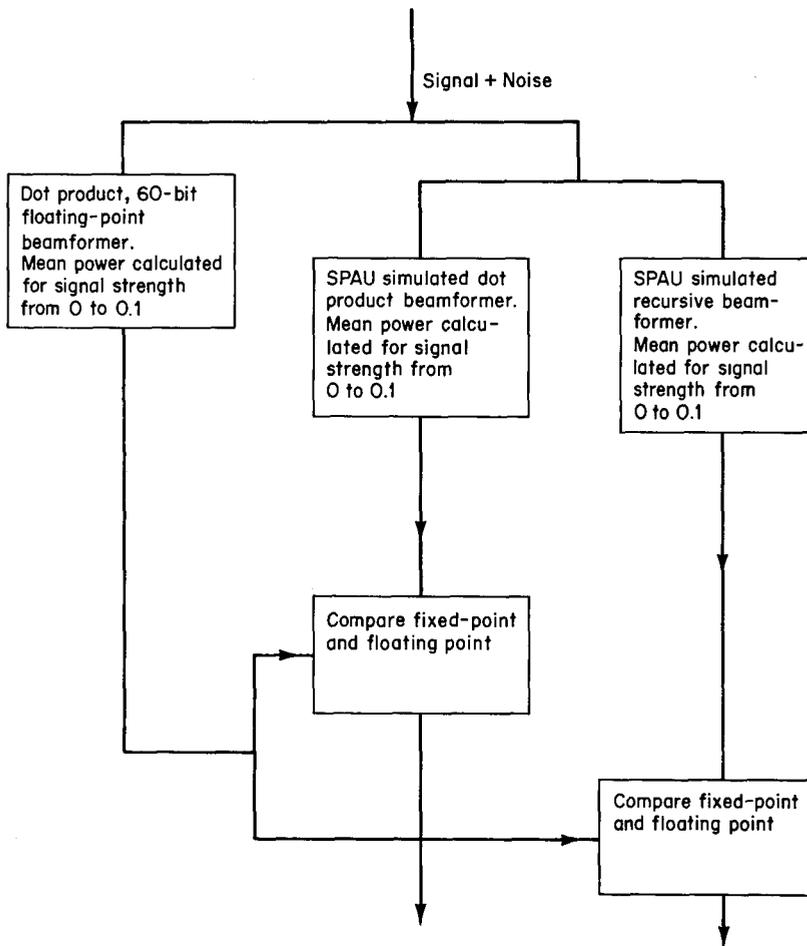


Fig. 6— Scheme for examining quantization for a low-strength signal

$$\begin{aligned}
 R(\theta) &= \sum_{k=1}^{34} \frac{1}{32} e^{2\pi i \underline{x}_k} e^{(-2\pi i k f d \sin \theta)/c} \\
 &= \frac{1}{32} \sum_{k=1}^{34} e^{2\pi i (\underline{x}_k - [dk/\lambda] \sin \theta)}
 \end{aligned}$$

where  $\underline{x}_k$  is the random variable with a uniform probability density  $f_{x_k}(x) = 1$  and  $x \in [0, 1]$ . Now the power  $P(\theta)$  is the squared complex magnitude of  $R(\theta)$  above. Let  $\underline{y}_k = \underline{x}_k - (dk/\lambda) \sin \theta$ . Then

$$\begin{aligned}
 P(\theta) &= \frac{1}{32^2} \sum_{k=1}^{34} e^{2\pi i \underline{y}_k} \sum_{n=1}^{34} e^{-2\pi i \underline{y}_n} \\
 &= \frac{1}{32^2} \left[ 34 + \sum_{\substack{k \neq j \\ k=1,34 \\ j=1,34}} e^{2\pi i \underline{y}_k} e^{-2\pi i \underline{y}_j} \right] \tag{4}
 \end{aligned}$$

$$E[e^{2\pi i \underline{y}_k} e^{-2\pi i \underline{y}_j}] = 0 \quad \forall k, j$$

since the random variables are independent and  $e^{\pm 2\pi i \underline{y}_k}$  is analytic on and inside the unit circle. Therefore,

$$\begin{aligned}
 E[P(\theta)] &= E \left\{ \frac{34}{32^2} \right\} \\
 &= \frac{34}{32^2} \\
 &= 0.03325,
 \end{aligned}$$

since a translation does not affect the expected value so long as  $\underline{x}_k$  ranges over its entire domain.

*Theoretical Derivation of the Variance Due to Noise*—The variance  $\sigma_{\underline{x}}^2$  is defined to be

$$\sigma_{\underline{x}}^2 = E\{(\underline{x} - E[\underline{x}])^2\} = E[\underline{x}^2] - E[\underline{x}]^2.$$

We must find  $E[P(\theta)^2]$ . From Eq. (4), we have

$$[P(\theta)]^2 = \frac{1}{32^4} \left[ 34 + \sum_{\substack{k \neq j \\ k=1,34 \\ j=1,34}} e^{2\pi i y_k} e^{-2\pi i y_j} \right]^2$$

where  $y_j$ ,  $i = j$ ,  $k$  is defined as before.

$$[P(\theta)]^2 = \frac{1}{32^4} \left[ 34^2 + 2 \cdot 34 \sum_{\substack{k \neq j \\ k=1,34 \\ j=1,34}} e^{2\pi i y_k} e^{-2\pi i y_j} \right. \\ \left. + \sum_{\substack{k \neq j \\ k=1,34 \\ j=1,34}} e^{2\pi i y_k} e^{-2\pi i y_j} \sum_{\substack{l \neq m \\ l=1,34 \\ m=1,34}} e^{2\pi i y_l} e^{-2\pi i y_m} \right].$$

As shown for Eq. (4), the expected value for the second term is 0.

Now consider the last expression. The terms in the product of the two summations are the product of four complex exponentials. As shown for Eq. (4), expressions of this form have an expected value of 0 except when  $k = m$  and  $j = l$  in which case, the expected value is 1. There are

$$P_{34,2} = \frac{34!}{32!}$$

terms whose expected value is 1. Therefore,

$$\sigma_x^2 = \frac{34^2}{32^4} + \frac{1}{32^4} \frac{34!}{32!} - \frac{34^2}{32^4} \\ = \frac{1}{32^4} (34.33).$$

Now we wish to find the sample variance which is  $\sigma^2 = \sigma_x^2 / N$  for uncorrelated data [12]:

$$\sigma^2 = \frac{34.33}{32^4} \cdot \frac{1}{5000} \\ = 0.214 \times 10^{-6}.$$

## Results and Comments

The mean and variance for the four steering directions are shown in tabular form in Appendix B, and the simulated mean and variance agree very well with those theoretically derived earlier. The mean power is plotted vs signal strength for the array steered at broadside in Fig. 7. The simulated results from the recursive algorithm and the ideal results are essentially the same with the resolution used for this curve.

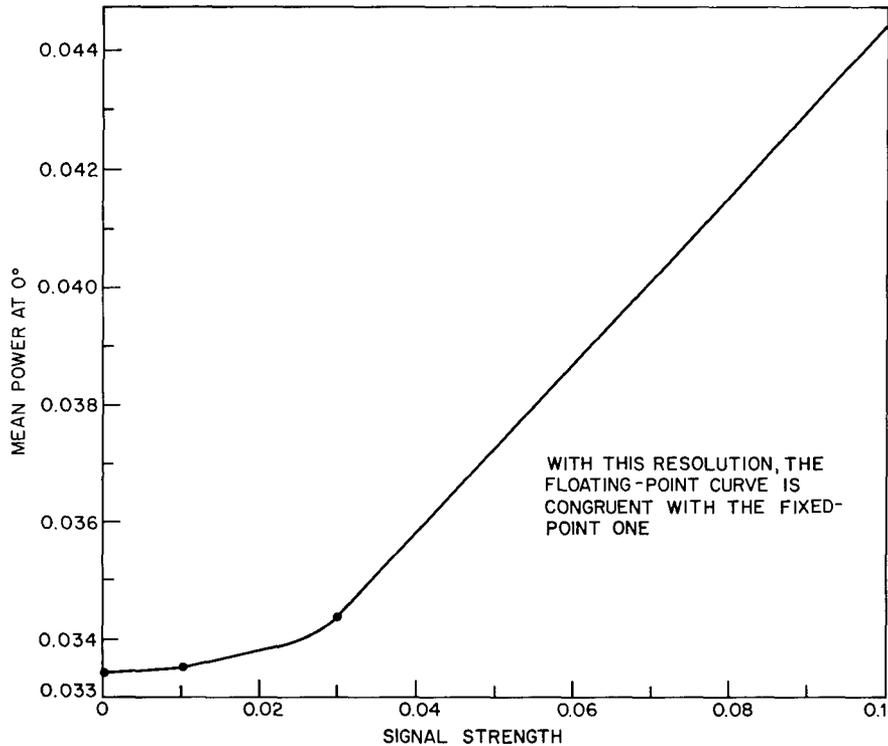


Fig. 7 — Signal strength vs mean power at  $0^\circ$ , the steering direction.

In Fig. 8, the section of the graphs with signal magnitude from 0 to 0.01 is magnified. As can be seen, the curve corresponding to the SPAU simulated recursive algorithm is within a standard deviation of the “ideal” curve. Contrary to expectation, the mean power increases slightly for decreasing signal strengths from 0.001 to 0 for the array steered at  $0^\circ$ . This happens because 5000 trials are not sufficient to ensure that the random numbers generated by the KRONOS random number function are uncorrelated. As an experiment to test this hypothesis, the random number generator was called 200,000 times and then the mean power calculated after 5000 trials for these signal amplitudes. The mean power decreased as it should.

For each steering angle, the mean power calculated by the SPAU simulation and the “ideal” floating-point mean power agree very well (to within a standard deviation of each other). This experiment gives confidence in the SPAU’s ability to beamform accurately.

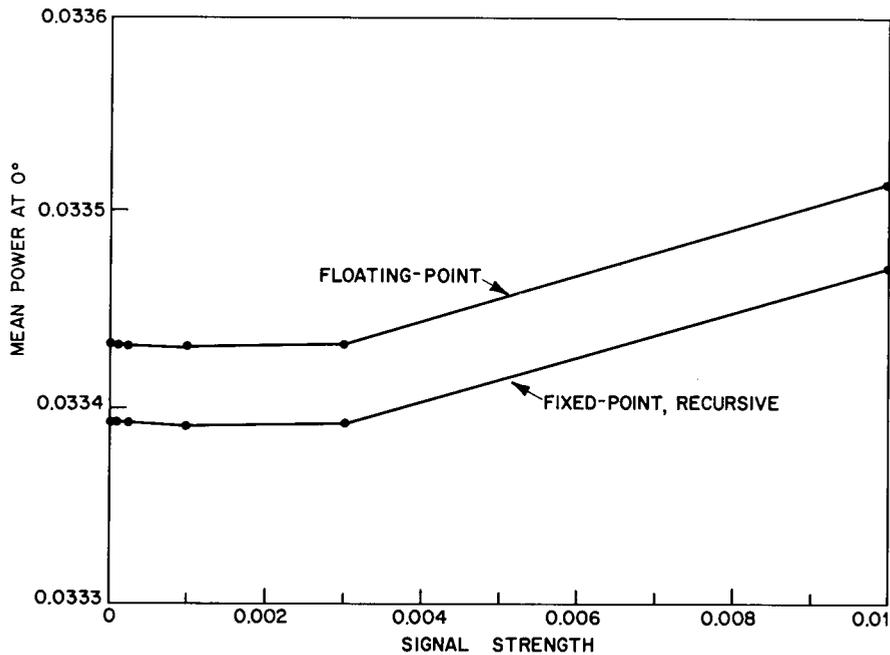


Fig. 8—Signal strength vs mean power at  $0^\circ$ , the steering direction (magnified)

## THE SPAU MACRO FOR FREQUENCY-DOMAIN BEAMFORMING

We must choose one algorithm for the SPAU macro, although the other algorithms will be available. Here, the relative merits of each algorithm will be reviewed and the macro microcoded.

### Pros and Cons for Each Algorithm

Let us first examine the FFT. In its favor, the FFT is the most accurate of the three algorithms since the input data do not have to be prescaled. The scaling is performed on blocks of data during various stages of execution; hence, this conditional block scaling puts out a bit of quantization noise when it is needed. The FFT is also very well suited to the SPAU's architecture, since the SPAU was designed with the FFT in mind. The FFT, as a beamforming algorithm, does have its drawbacks. For the array that is referenced in this study, it is necessary to use at least a 128-point FFT and up to a 2048-point FFT to obtain the required resolution for different frequencies [3]. These FFTs require more time for execution than the dot product or the recursive Goertzel algorithm. Steering directions are inflexible. When the FFT is used to beamform in the frequency domain,  $R(Nf(d \sin \theta)/c)$ , where  $N$  is the length of the FFT, is calculated, and  $R(\theta)$  must be obtained. In other words, the output block must be searched to obtain the desired data point in order to find a subscript  $k$  such that  $\theta = \sin^{-1}(kc/Nfd)$ . For a given frequency  $f$ , element spacing  $d$ , and FFT length  $N$ , a list of steering angles  $\theta$  can be obtained.

Now let us consider the dot product. The input to this algorithm must be scaled by  $K$ , the number of array elements, to prevent overflow. Hence, at the initiation of the

computation  $\log_2 K$  bits are lost, the whole word length is not used, and the quantization noise accumulates as the calculation progresses and is carried throughout. For these reasons, the dot product is not as accurate as the FFT. For each steering direction,  $K$  steering phasors must be available for the execution of the algorithm. Beamforming is generally involved in real-time applications where time is critical, and the required steering phasors must be precomputed and stored. Suppose that we wish to examine 512 frequencies and 32 steering directions. A steering phasor is required for each hydrophone element; hence, 557,056 sine-cosine pairs must be generated and stored. To contemplate storing this large number of steering phasors is unrealistic because storage is the most expensive component of the SPE. Unless it is possible to generate the phasors as they are needed, it would not be advisable to use the dot product and because the SPE is to be used for real-time applications, it is unlikely that the phasors could be generated during processing time.

Now let us look at the recursive algorithm. Both the recursive and the dot product require the input data to be scaled by  $K$ , the number of array elements. The accuracy of both algorithms are essentially the same. Both algorithms require the same number of complex and real operations. One important difference between the dot product and the recursive algorithm is that the recursive algorithm requires only one steering phasor for each beam direction and frequency. For 512 frequencies and 32 steering directions, only 16,384 sine-cosine pairs are required—storage requirements decrease by a factor of  $K$  over the dot product. Hence, the recursive algorithm is our candidate for the SPAU beamforming macro.

### SPAU Macro Coded in ANIMIL

The SPAU has a microprogramming language ANIMIL [13,14] in which we coded the recursive algorithm. While the other algorithms will be available, the recursive algorithm will be the preferred one.

The steering phasors are generated sometime prior to the beamforming computation and are stored in the SPAU Coefficient Store (CFS). The CFS has 1025 32-bit words which are read-only and 1023 64-bit words which are both read and write. Sine-cosine pairs whose arguments range from 0 to  $\pi/2$  in increments of  $\pi/2048$  reside in the first 1025 words. The remaining 1023 64-bit words contain coefficients such as steering phasors or filter coefficients. There are four 16-bit registers that are loaded from the CFS.

The hydrophone data are stored in SPE buffer memory. They are brought into the  $X$  or  $Y$  local stores of the SPAU via two 32-bit-wide data channels. The inputs to the SPAU's four 150-ns (one cycle) multipliers can be selected from the  $X$  or  $Y$  stores, from the  $Z$  registers, and from adders. The inputs to the SPAU's four adders can be selected from multiplier outputs, other adder outputs, and the local stores. Adders 2 and 4 can choose as inputs the results of adders 1 and 3, respectively, and both additions are executed in one cycle. For example, at the beginning of a cycle, inputs are selected for adder 1, the result is obtained and fed into adder 2 as input, and the result is produced by the end of the cycle.

Figure 9 shows how the hardware in the Arithmetic Section of the SPAU is used for the recursive beamforming algorithm. The 32-bit  $X$  register ( $X1$  contains the real 16 bits and  $X2$ , the imaginary 16 bits) is initialized to the first piece of complex hydrophone data, and throughout the remainder of the calculation it serves as a receptacle for the complex feedback term that is then fed through the multipliers. The hydrophone data are brought into the  $Y$  store where they are added to the product of the feedback term and the steering phasor located in the  $Z$  registers. By initializing the  $X$  registers with the first complex data point, the loop is required to be executed 33 times instead of 34. Now it takes two cycles to calculate the feedback term; that is, the multipliers are idle every other cycle. To use the hardware more efficiently, two beams can be formed at the same time; while the multipliers are occupied with the first set of hydrophone data, the adders are busy with the second set, and vice versa.

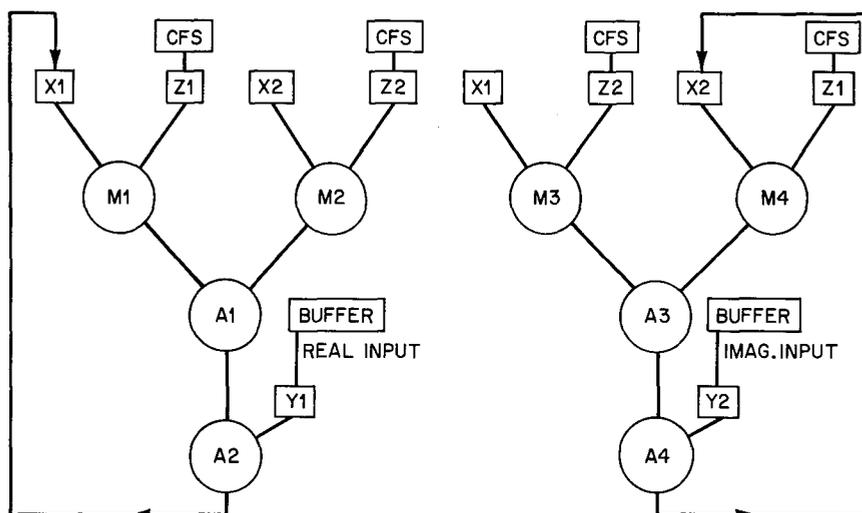


Fig. 9—Use of SPAU facilities for the recursive algorithm

The source listing for SPAU recursive beamforming macro coded in ANIMIL follows. The 16-bit  $W$  local stores and adders 5 through 7 are located in the Address Generator section of the SPAU. In this macro, A5 and A6 are used to compute buffer addresses and A7 is used to calculate the appropriate CFS address. BARA contains buffer A address; BARB, buffer B address; and RAR, the CFS address. The SPAU has three counters (CTRI, CTRJ, CTRK); these counters are decremented and tested for the zero condition and can thus be used to control looping. The SPAU control word also has a literal field that can be used to initialize counters, address registers, or data registers.

## CONCLUSION

After a careful investigation of time-domain and frequency-domain beamforming, it was decided to use frequency-domain array processing for the SPE/PCPS program because frequency-domain beamforming conforms more closely to efficient use of SPE resources. Three frequency-domain algorithms—the dot product, the recursive algorithm, and the FFT—were studied to determine the most efficient and accurate beamforming algorithm

J. FROSCHER

\*\*\* ANIMIL VERSION 1.0 74/10/29 10.06.15 \*\*\*

ENTER SOURCE FILE NAME, OBJECT FILE NAME, AND OPTIONAL PARAMETERS  
 ? BEAM, BEAMX, SO, L=72

SOURCE PROGRAM AND ERROR LISTING

ADDRESS	MICROINSTRUCTION
0	"THIS IS A SPAU MICROPROGRAM TO FORM A BEAM IN EACH OF TWO
0	DIRECTIONS VIA THE RECURSIVE ALGORITHM. THE SENSOR DATA
0	FOR THE FIRST BEAM DIRECTION IS IN BSM 0 AND THE DATA FOR
0	THE SECOND DIRECTION IN IN BSM 1. THE STEERING PHASORS ARE
0	STORED IN THE WRITABLE PORTION OF THE ROM. THIS ARRAY HAS
0	34 ELEMENTS EACH SPACED 15 FEET APART. IT IS ASSUMED THAT
0	THE INPUT DATA HAS ALREADY BEEN SCALED BY 32."
0	A5: BARRA=W(1);\$
1	A5: INCA=W(3);\$
2	A6: BARB=W(2); A7: RAR=LIT; LIT=1025;\$
3	A6: INCB=W(3);\$
4	CTR1=W(4);\$
5	CTRJ=W(4);\$
6	A7: INCR=LIT; LIT=1; X(1)=BUFA; A5: BARA=BARA+INCA;\$
7	Z=ROM; X(2)=BUFB; A7: RAR=RAR+INCR;\$
8	Z=ROM; Y(1)=BUFA;
8	M1:X1(1)*Z1;
8	M2:X2(1)*Z2;
8	M3:X1(1)*Z2;
8	M4:X2(1)*Z1;
8	A6: BARB=BARB+INCB; A7: RAR=RAR-INCR;\$
9	LOOP# Z=ROM; Y(2)=BUFB;
9	M1:X1(2)*Z1; M2:X2(2)*Z2; M3:X1(2)*Z2; M4:X2(2)*Z1;
9	A1:P1-P2; A2:X1(1)=Y1(1)+A1; A3:P3+P4; A4:X2(1)=Y2(1)+A3;
9	A5:BARA=BARA+INCA; A7:RAR=RAR+INCR;
9	IF CTRJ THEN GO TO OUT; DECJ;\$
10	Z=ROM; Y(1)=BUFA;
10	M1:X1(1)*Z1; M2:X2(1)*Z2; M3:X1(1)*Z2; M4:X2(1)*Z1;
10	A1:P1-P2; A2:X1(2)=Y1(2)+A1; A3:P3+P4; A4:X2(2)=Y2(2)+A3;
10	A6:BARB=BARB+INCB; A7:RAR=RAR-INCR;
10	IF NOT CTR1 THEN GO TO LOOP; DECI;\$
11	OUT#
11	A1:P1-P2; A2:X1(2)=Y1(2)+A1; A3:P3+P4; A4:X2(2)=Y2(2)+A3;
11	BUFA=X(1);\$
12	BUFB=X(2); INTERRUPT;\$

NO TRANSLATION ERRORS FOUND

END.

for the SPAU. After performing two quantization experiments, it was determined that the FFT was the most accurate algorithm and that the dot product was essentially as accurate as the recursive algorithm and vice versa. Unfortunately, the FFT does not permit the flexible choice of a steering direction—only certain steering directions could be obtained. Also, the processing time for the execution of the FFT was longer than for either of the other algorithms because a longer FFT had to be used to obtain the needed resolution. The dot product and the recursive algorithm used the same number of complex and real operations. The dot product, however, required the generation and storage of a large number of steering phasors—34 times as many as for the recursive algorithm (for this array). Hence, a consideration of the tradeoffs for the three algorithms determined that the recursive Goertzel algorithm is the most promising beamforming macro for the SPE.

## ACKNOWLEDGMENT

The author wishes to thank R. Cruse, J. L. Schilling, J. Spieser, and W. Smith for helpful discussions on acoustic array processing.

## REFERENCES

1. W. R. Smith and J. P. Ihnat, "Signal Processing Element User's Reference Manual," NRL Report 7488, Sept. 5, 1972.
2. W. R. Smith and H. H. Smith, "Signal Processing Element Functional Description: Part 2, Preliminary Signal Processing Arithmetic Unit," NRL Memorandum Report 2522, Oct. 1972.
3. J. M. Speiser, "Working Notes on Line Array Processing with PCPS," NUC TN 1121, Aug. 6, 1973.
4. T. Elbert, "Adaptive Signal Processing for Point Sensor Arrays," TR 70-34, May 1970, General Motors Corp., AC Electronics-Defense Research Laboratories.
5. R. L. Pritchard, "Optimum Directivity Patterns for Linear Point Arrays," *J. Acoust. Soc. Amer.* 25, 879-891 (1953).
6. J. Thompson and B. J. Goldstone, "Digital Signal Processing," *Computer Design* 8(No. 5), 44-49 (May 1960).
7. J. W. Cooley and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Math. Comput.* 19, 297-301 (1965).
8. L. B. W. Jolley, *Summation of Series*, 2d ed., Dover Publications, New York, 1961.
9. J. N. Froscher, "Effects of Finite Register Length in the Signal Processing Arithmetic Unit of the AN/UYK-17," NRL Report 7732, June 12, 1974.
10. B. Gold and C. M. Rader, *Digital Processing of Signals*, McGraw-Hill, New York, 1969.
11. J. R. Williams, "Fast Beam-Forming Algorithm," *J. Acoust. Soc. Amer.* 44, 1454-1455 (1968).

J. FROSCHER

12. W. B. Davenport, Jr., and W. L. Root, *An Introduction to the Theory of Random Signals and Noise*, McGraw-Hill, New York, 1958.
13. T. Rauscher, J. Roberts, and W. D. Elliott, "AN/UYK-17(XB-1)(V) Microprogramming Support Software," NRL Report 7777, in preparation.
14. W. R. Smith, J. P. Ihnat, H. H. Smith, N. M. Head, Jr., E. Freeman, Y. S. Wu, and B. Wald, "AN/UYK-17(XB-1)(V) Signal Processing Element Architecture," NRL Report 7704, June 7, 1974.

## Appendix A

### EXPERIMENTAL RESULTS FOR THE CASE OF A STRONG INTERFERING SIGNAL

Experiments in which the steering direction, the interfering signal direction, and the relative amplitude of the signal in the steering direction were varied, were performed for all three algorithms for both rounded and truncated arithmetic. The interfering amplitude remained constant at 0.5. The array was kept at a frequency of 125 Hz and was assumed to be uniformly weighted. All responses in Tables A1-A14 are magnitudes. Figures A1 through A5 depict the discrepancies between the floating-point results and the fixed-point simulator results.

Table A1  
Theoretical Response Due to Interfering Signal Alone = 0.052195742  
(Beam steered at 0°; interfering signal at 5°.)

Amplitude at 0°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling)
Truncated				
0.5	0.58200610	0.58039041	0.58039041	0.58170345
0.4	0.47578730	0.47454105	0.47438025	0.47547263
0.3	0.36958643	0.36863909	0.36843207	0.36925945
0.2	0.26342519	0.26238837	0.26209744	0.26314328
0.1	0.15738379	0.15664014	0.15625000	0.15702931
0.0	0.052197763	0.051229988	0.051229988	0.051822262
Rounded				
0.5	0.58200610	0.58138859	0.58133610	0.58188704
0.4	0.47578730	0.47540844	0.47540844	0.47566514
0.3	0.36958643	0.36942470	0.36942470	0.3694600
0.2	0.26342519	0.26314328	0.26314328	0.26331718
0.1	0.15738379	0.15732055	0.15732055	0.15732005
0.0	0.052197763	0.051822262	0.051822262	0.052125875

J. FROSCHER

Table A2  
 Theoretical Response Due to Interfering Signal Alone = 0.0014611491  
 (Beam steered at 0°; interfering signal at 9°.)

Amplitude at 0°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling)
Truncated				
0.5	0.52981904	0.52822554	0.52825442	0.52949503
0.4	0.42356906	0.42248943	0.42227267	0.42328324
0.3	0.31731909	0.31643036	0.31614090	0.31700849
0.2	0.21106916	0.21014028	0.20984963	0.21079277
0.1	0.10481937	0.10423175	0.10379164	0.10452413
0.0	0.0014605774	0	0	0
Rounded				
0.5	0.52981904	0.52926444	0.52926444	0.52969672
0.4	0.42356906	0.42331929	0.42331929	0.42346344
0.3	0.31731909	0.31715286	0.31715286	0.31720097
0.2	0.21106916	0.21086515	0.21086515	0.21093750
0.1	0.10481937	0.10481569	0.10481569	0.10467001
0.0	0.0014605774	0	0	0

Table A3  
 Theoretical Response Due to Interfering Signal Alone = 0.068542188  
 (Beam steered at 0°; interfering signal at 11°.)

Amplitude at 0°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling)
Truncated				
0.5	0.56362338	0.56212010	0.56206580	0.56350280
0.4	0.45817072	0.45701456	0.45681418	0.45784849
0.3	0.35319630	0.35234288	0.35199626	0.35290542
0.2	0.24930498	0.24840802	0.24803919	0.24908279
0.1	0.14878320	0.14833467	0.14781944	0.14854026
0.0	0.068542200	0.068331468	0.068107796	0.068331468
Rounded				
0.5	0.56362338	0.56309647	0.56298807	0.56350280
0.4	0.45817072	0.45784849	0.45784849	0.45801510
0.3	0.35319630	0.35299189	0.35299189	0.35312154
0.2	0.24930498	0.24908279	0.24908279	0.24926650
0.1	0.14878320	0.14874557	0.14874557	0.14874557
0.0	0.068542200	0.068331468	0.068331468	0.068554409

Table A4  
 Theoretical Response Due to Interfering Signal Alone = 0.10937610  
 (Beam steered at 3°; interfering signal at 9°.)

Amplitude at 3°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 2.99°
Truncated				
0.5	0.60492088	0.60333546	0.60469961	0.60520408
0.4	0.49997977	0.49865542	0.49996948	0.50027458
0.3	0.39573778	0.39439586	0.39567054	0.39605600
0.2	0.29294221	0.29158528	0.29283851	0.29330710
0.1	0.19390723	0.19279629	0.19382244	0.19437274
0.05	0.14818653	0.14699131	0.14812879	0.14864295
0.02	0.12362913	0.12265874	0.12340288	0.12401959
0.01	0.11623552	0.11521782	0.11614116	0.11666550
0.00	0.10937556	0.10825318	0.10909563	0.10979273
Rounded				
0.5	0.60492088	0.60439673	0.60437149	0.60515365
0.4	0.49997977	0.49957257	0.49966419	0.50024408
0.3	0.39573778	0.39532331	0.39547767	0.39597893
0.2	0.29294221	0.29268215	0.29263001	0.29330710
0.1	0.19390723	0.19366493	0.19374370	0.19429422
0.05	0.14818653	0.14792263	0.14802574	0.14864295
0.02	0.12362913	0.12340288	0.12352647	0.12414257
0.01	0.11623552	0.11587810	0.11614116	0.11666550
0.00	0.10937556	0.10937500	0.10923540	0.10993162

Table A5  
 Theoretical Response Due to Interfering Signal Alone = 0.044160619  
 (Beam steered at 3°; interfering signal at 12°.)

Amplitude at 3°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 2.99°
Truncated				
0.5	0.52695444	0.52538699	0.52680818	0.52782096
0.4	0.42070469	0.41944468	0.42093356	0.42151316
0.3	0.31445512	0.31308539	0.31444706	0.31531931
0.2	0.20820598	0.20706810	0.20824380	0.20904825
0.1	0.10195862	0.10095975	0.10216169	0.10275739
0.05	0.048839258	0.047199398	0.048789047	0.049410588
0.02	0.016984575	0.01562500	0.016572815	0.017469281
0.00	0.0044178217	0	0	0
Rounded				
0.5	0.52695444	0.52643150	0.52625756	0.52776314
0.4	0.42070469	0.42046204	0.42031686	0.42151316
0.3	0.31445512	0.31430145	0.31425290	0.31527092
0.2	0.208220598	0.20817051	0.20809720	0.20904825
0.1	0.10195862	0.10186254	0.10186254	0.10275739
0.05	0.048839258	0.049100801	0.048475288	0.049708446
0.02	0.016984575	0.016572815	0.016572815	0.017469281
0.00	0.0044178217	0.0055242717	0	0

Table A6  
 Theoretical Response Due to Interfering Signal Alone = 0.068170501  
 (Beam steered at 3°; interfering signal at 14°.)

Amplitude at 3°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 2.99°
Truncated				
0.5	0.56616246	0.56463895	0.56604245	0.56620417
0.4	0.46065728	0.45924610	0.46054008	0.46073883
0.3	0.35559592	0.35432939	0.35566186	0.35570476
0.2	0.25153510	0.25032594	0.25152125	0.25164255
0.1	0.15056389	0.14946193	0.15058074	0.15058074
0.05	0.10409644	0.10275739	0.10364452	0.10423175
0.02	0.080283220	0.079288215	0.080054303	0.080434610
0.00	0.068170269	0.067205666	0.067883387	0.068107796
Rounded				
0.5	0.56616246	0.56571887	0.56561097	0.56615027
0.4	0.46065728	0.46034125	0.46014232	0.46067259
0.3	0.35559592	0.35540436	0.35336142	0.35561896
0.2	0.25153510	0.25127847	0.25133919	0.25158191
0.1	0.15056389	0.15058074	0.15037794	0.15068204
0.05	0.10409644	0.10379164	0.10379164	0.10408525
0.02	0.080283220	0.080244682	0.080244682	0.080434610
0.00	0.068170269	0.068107796	0.068107796	0.068331468

Table A7  
 Theoretical Response Due to Interfering Signal Alone = 0.0088881422  
 (Beam steered at 15°; interfering signal at 6°.)

Amplitude at 15°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 15.1°
Truncated				
0.5	0.52272415	0.52134293	0.52221172	0.52758964
0.4	0.41647567	0.41520332	0.41615773	0.42140454
0.3	0.31022824	0.30921121	0.30995054	0.31527092
0.2	0.20398347	0.20297470	0.20379996	0.20904825
0.1	0.097750098	0.096635288	0.097421593	0.10275739
0.05	0.044663078	0.043498159	0.044194174	0.049410588
0.02	0.012961134	0.011048543	0.012352647	0.017469281
0.00	0.0088899290	0.0078125000	0.0078125000	0
Rounded				
0.5	0.52272415	0.52229937	0.52206560	0.52761856
0.4	0.41647567	0.41623105	0.41608439	0.42147695
0.3	0.31022824	0.30999976	0.30999976	0.31527092
0.2	0.20398347	0.20379996	0.20379996	0.20912123
0.1	0.097750098	0.097578094	0.097578094	0.10305395
0.05	0.044663078	0.044879396	0.044879396	0.050024408
0.02	0.012961134	0.013531647	0.012352647	0.018321937
0.00	0.0088899290	0.0078125000	0.0095683193	0

Table A8  
 Theoretical Response Due to Interfering Signal Alone = 0.030320258  
 (Beam steered at 15°; interfering signal at 20°.)

Amplitude at 15°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 15.1°
Truncated				
0.5	0.56144454	0.56013500	0.56103324	0.55134889
0.4	0.45519619	0.45399969	0.45470485	0.44517542
0.3	0.34894885	0.34780983	0.34859862	0.33905710
0.2	0.24270382	0.24168294	0.24250232	0.23287279
0.1	0.13646655	0.13554181	0.13621559	0.12657697
0.05	0.083360074	0.082494969	0.083231546	0.073495655
0.02	0.051514532	0.050630787	0.051229988	0.041707337
0.00	0.030318587	0.029231698	0.030257682	0.019918045
Rounded				
0.5	0.56144454	0.56108764	0.56089724	0.55134889
0.4	0.45519619	0.45483916	0.45473851	0.44520969
0.3	0.34894885	0.34864239	0.34851106	0.33905710
0.2	0.24270382	0.24250232	0.24237644	0.23287279
0.1	0.13646655	0.13643945	0.13655124	0.12669746
0.05	0.083360074	0.083414674	0.083414674	0.073702978
0.02	0.051514532	0.051229988	0.051526976	0.041707337
0.00	0.030318587	0.030257682	0.030257682	0.020669932

Table A9  
 Theoretical Response Due to Interfering Signal Alone = 0.11458246  
 (Beam steered at 15°; interfering signal at 22°.)

Amplitude at 15°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 15.1°
Truncated				
0.5	0.56642840	0.56504416	0.56598853	0.56633890
0.4	0.46275533	0.46136765	0.46225976	0.46278760
0.3	0.36058561	0.35933254	0.36022318	0.36077343
0.2	0.26168606	0.26046224	0.26122271	0.26198098
0.1	0.17180028	0.17035908	0.17134150	0.17231832
0.05	0.13601471	0.13486466	0.13576677	0.13666294
0.02	0.12083119	0.11976336	0.12065192	0.12153398
0.00	0.11458272	0.11334858	0.11415343	0.11535018
Rounded				
0.5	0.56642840	0.56606940	0.56588068	0.56633890
0.4	0.46275533	0.46229276	0.46212770	0.46282057
0.3	0.36058561	0.36026554	0.36018082	0.36077343
0.2	0.26168606	0.26139789	0.26133951	0.26203922
0.1	0.17180028	0.17160846	0.17151952	0.17240685
0.05	0.13601471	0.13587912	0.13576677	0.13677454
0.02	0.12083119	0.12065192	0.12077832	0.12153398
0.00	0.11458272	0.11455374	0.11442146	0.11548239

J. FROSCHER

Table A10  
Theoretical Response Due to Interfering Signal Alone = 0.031601997  
(Beam steered at 15°; interfering signal at 24°.)

Amplitude at 15°	"Ideal" Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 15.1°
Truncated				
0.5	0.50427847	0.50304250	0.50373968	0.49902248
0.4	0.39809663	0.39709486	0.39774756	0.39284525
0.3	0.29196440	0.29116633	0.29168992	0.28694324
0.2	0.18596670	0.18487749	0.18578316	0.18112536
0.1	0.080635882	0.079480429	0.080244682	0.076146831
0.05	0.030466500	0.029231698	0.029749114	0.027621359
0.02	0.017123668	0.015625000	0.016572815	0.021395412
0.00	0.031603815	0.031734525	0.031250000	0.037057941
Rounded				
0.5	0.50427847	0.50386083	0.50389111	0.49905306
0.4	0.39809663	0.39790099	0.39782428	0.39300059
0.3	0.29196440	0.29179452	0.29163760	0.28699642
0.2	0.18596670	0.18586527	0.18594735	0.18120959
0.1	0.080635882	0.080624092	0.080434610	0.076546554
0.05	0.030466500	0.030257682	0.030257682	0.028168369
0.02	0.017123668	0.017469281	0.017469281	0.022097087
0.00	0.031603815	0.031734525	0.031734525	0.037057941

Table A11  
Theoretical Response Due to Interfering Signal Alone = 0.0044093214  
(Beam steered at 45°; interfering signal at 39°.)

Amplitude at 45°	"Ideal" Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 45.1°
Truncated				
0.5	0.52686931	0.52628656	0.52611257	0.53482825
0.4	0.42061937	0.42067973	0.42035316	0.42865643
0.3	0.31436948	0.31430145	0.31415577	0.32240172
0.2	0.20811970	0.20802386	0.20780369	0.21622426
0.1	0.10187037	0.10171263	0.10171263	0.10993162
0.05	0.048746795	0.048789047	0.048475288	0.056875859
0.02	0.016876975	0.016572815	0.016572815	0.024705294
0.00	0.0044113533	0	0	0
Rounded				
0.5	0.52686931	0.52637353	0.52611257	0.53491364
0.4	0.42061937	0.42064346	0.42049833	0.42869203
0.3	0.31436948	0.31425290	0.31415577	0.32249637
0.2	0.20811970	0.20802386	0.20795049	0.21622426
0.1	0.10187037	0.00186254	0.10171263	0.11007033
0.05	0.048746795	0.048789047	0.048789047	0.056875859
0.02	0.016876975	0.016572815	0.017469281	0.025315393
0.00	0.0044113533	0.0055242717	0.0055242717	0.0055242717

Table A12  
 Theoretical Response Due to Interfering Signal Alone = 0.016235393  
 (Beam steered at 45°; interfering signal at 52°.)

Amplitude at 45°	"Ideal" Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 45.1°
Truncated				
0.5	0.54748326	0.54698660	0.54698660	0.53928890
0.4	0.44123326	0.44090472	0.44073165	0.43301270
0.3	0.33498326	0.33484561	0.33484561	0.32682032
0.2	0.22873326	0.22850728	0.22857404	0.22062533
0.1	0.12248327	0.12253427	0.12240968	0.11428702
0.05	0.069358283	0.069218926	0.068998132	0.061267137
0.02	0.037483305	0.037057941	0.037467434	0.029231698
0.00	0.016233369	0.015625000	0.015625000	0.0078125000
Rounded				
0.5	0.54748326	0.54698660	0.54698660	0.53931720
0.4	0.44123326	0.44090472	0.44083550	0.43315363
0.3	0.33498326	0.33480004	0.33480004	0.32691369
0.2	0.22873326	0.22864079	0.22857404	0.22076361
0.1	0.12248327	0.12240968	0.12253427	0.11455374
0.05	0.069358283	0.068998132	0.069439019	0.061267137
0.02	0.037483305	0.037467434	0.037467434	0.029749114
0.00	0.016233369	0.015625000	0.016572815	0.0078125000

Table A13  
 Theoretical Response Due to Interfering Signal Alone = 0.11573950  
 (Beam steered at 45°; interfering signal at 55°.)

Amplitude at 45°	"Ideal" Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 45.1°
Truncated				
0.5	0.58183041	0.58125735	0.58123110	0.58141484
0.4	0.47785372	0.47717046	0.47710650	0.47745817
0.3	0.37515310	0.37467434	0.37471506	0.37479649
0.2	0.27516114	0.27460687	0.27471798	0.27488456
0.1	0.18238854	0.18163012	0.18204969	0.18213349
0.05	0.14320733	0.14213598	0.14288548	0.14288548
0.02	0.12475323	0.12364994	0.12426542	0.12451076
0.00	0.11573949	0.11442046	0.11535018	0.11548239
Rounded				
0.5	0.58183041	0.58120485	0.58117859	0.58149356
0.4	0.47785372	0.47736229	0.47745817	0.47755404
0.3	0.37515310	0.37500000	0.37491861	0.37487791
0.2	0.27516114	0.27494007	0.27494007	0.27494007
0.1	0.18238854	0.18213349	0.18221725	0.18221725
0.05	0.14320733	0.14299223	0.14288548	0.14309890
0.02	0.12475323	0.12463325	0.12451076	0.12463325
0.00	0.11573949	0.11548239	0.11561444	0.11561444

Table A14  
 Theoretical Response Due to Interfering Signal Alone = 0.0069062934  
 (Beam steered at 45°; interfering signal at 60°.)

Amplitude at 45°	“Ideal” Floating Dot	Fixed Dot	Fixed Recursive	Fixed FFT (Conditional Scaling) THI = 45.1°
Truncated				
0.5	0.53811670	0.53775884	0.53747502	0.53368582
0.4	0.43186682	0.43156550	0.43142405	0.42755151
0.3	0.32561702	0.32555729	0.32546354	0.32131132
0.2	0.21936741	0.21930733	0.21902884	0.21509219
0.1	0.11311853	0.11294400	0.11294400	0.10881553
0.05	0.059995573	0.059754135	0.059754135	0.055518244
0.02	0.028125508	0.027621359	0.027621359	0.023437500
0.00	0.0069041587	0.0055242717	0.0055242717	0
Rounded				
0.5	0.5311670	0.53753179	0.53761695	0.53380018
0.4	0.43186682	0.43153014	0.43149478	0.42762288
0.3	0.32561702	0.32541665	0.32546354	0.32145376
0.2	0.21936741	0.21916813	0.21923774	0.21516311
0.1	0.11311853	0.11307902	0.11307902	0.10895567
0.05	0.059995573	0.060008951	0.060008951	0.055792410
0.02	0.028125508	0.027621359	0.028168369	0.024079742
0.00	0.0069041587	0.0078125000	0.0078125000	0

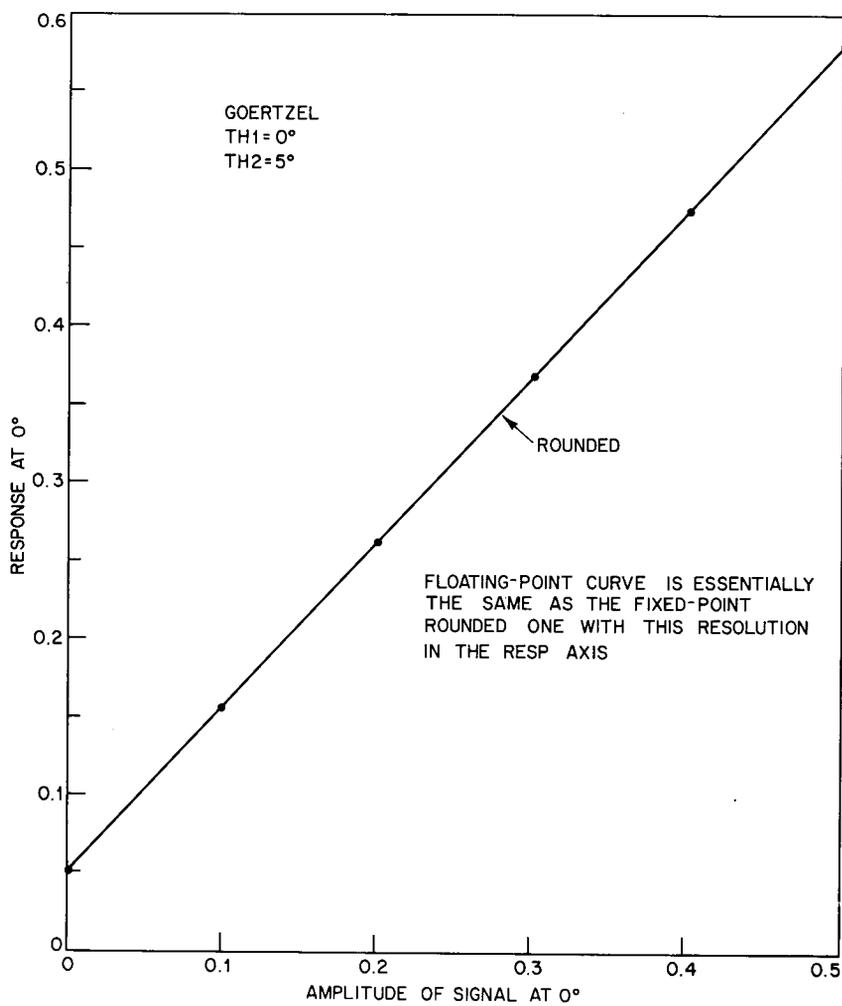


Fig. A1— Signal strength at 0°, the steering direction, vs magnitude of response with the interfering signal at 5°.

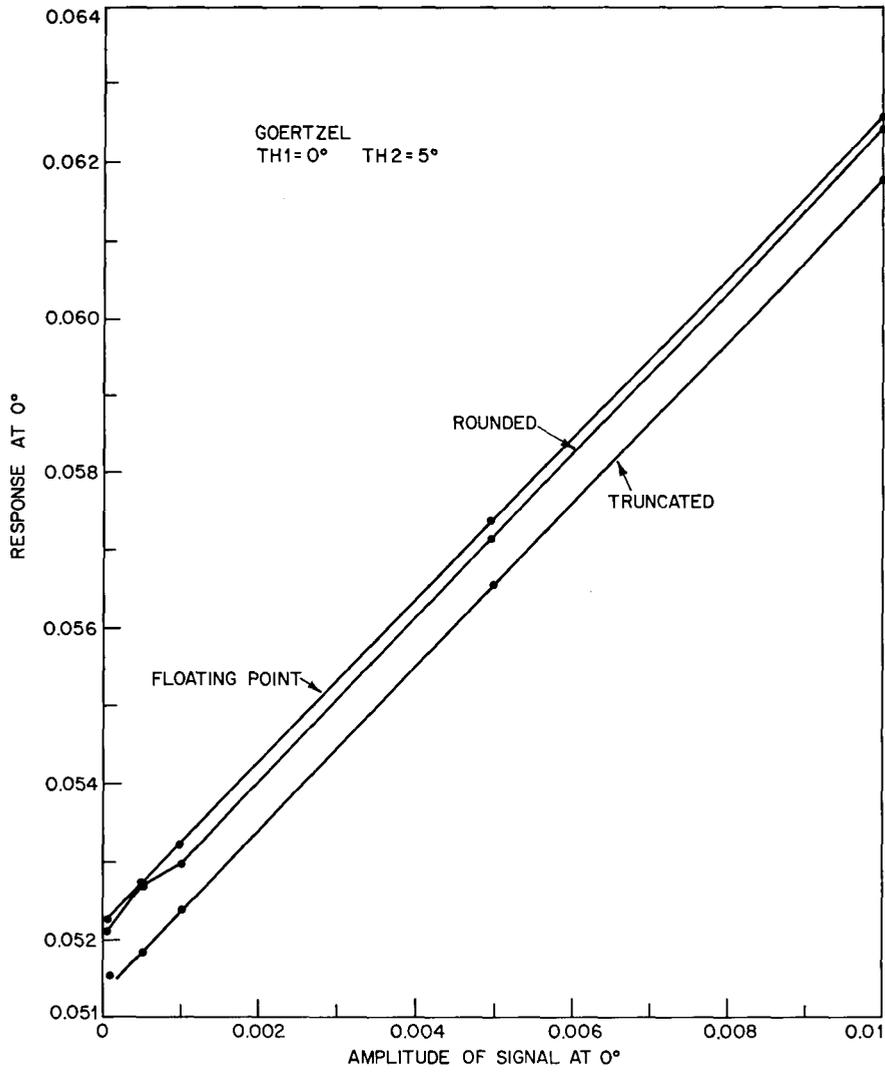


Fig. A2—Signal strength at 0°, the steering direction, vs magnitude of response with the interfering signal at 5° (magnified)

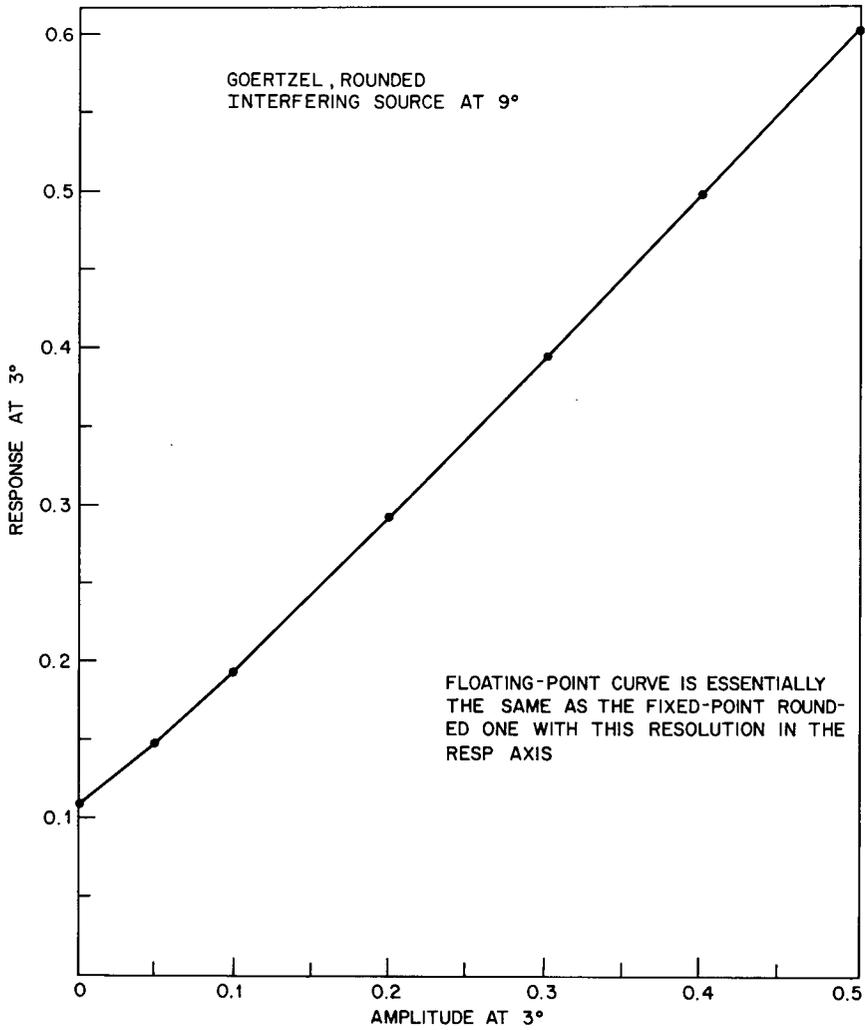


Fig. A3—Signal strength at 3°, the steering direction, vs magnitude of response with the interfering signal at 9°

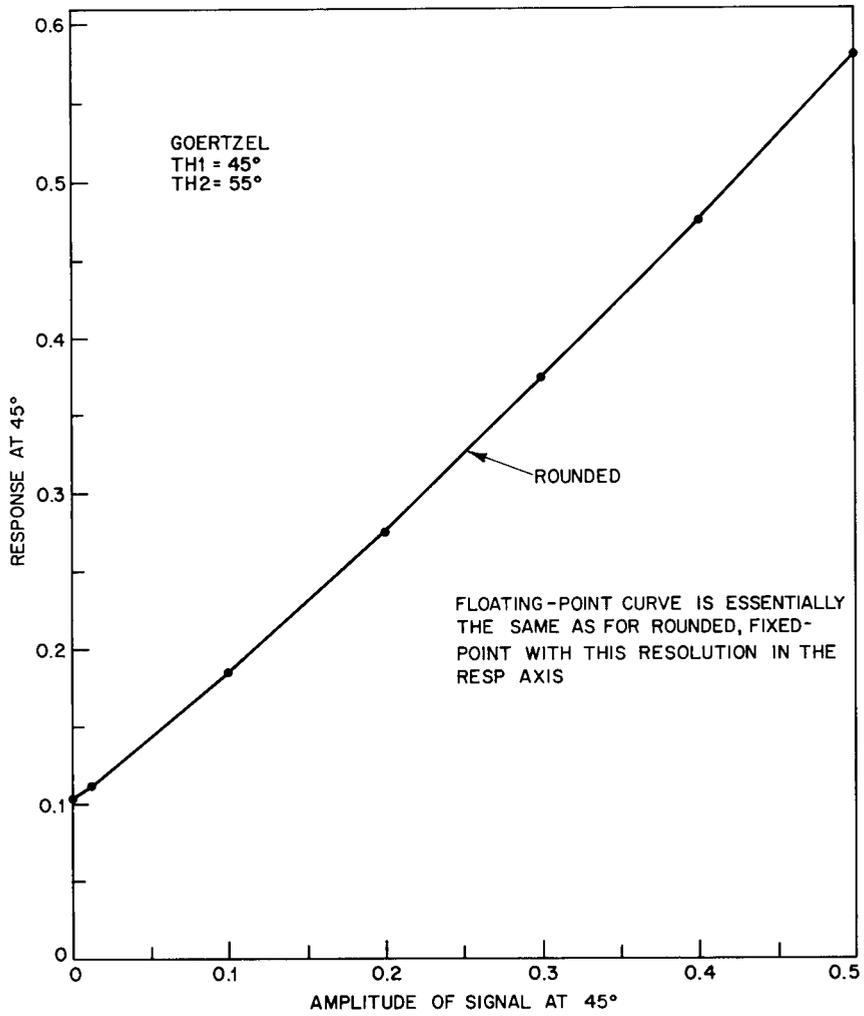


Fig. A4—Signal strength at 45°, the steering direction, vs magnitude of response with interfering signal at 55°.

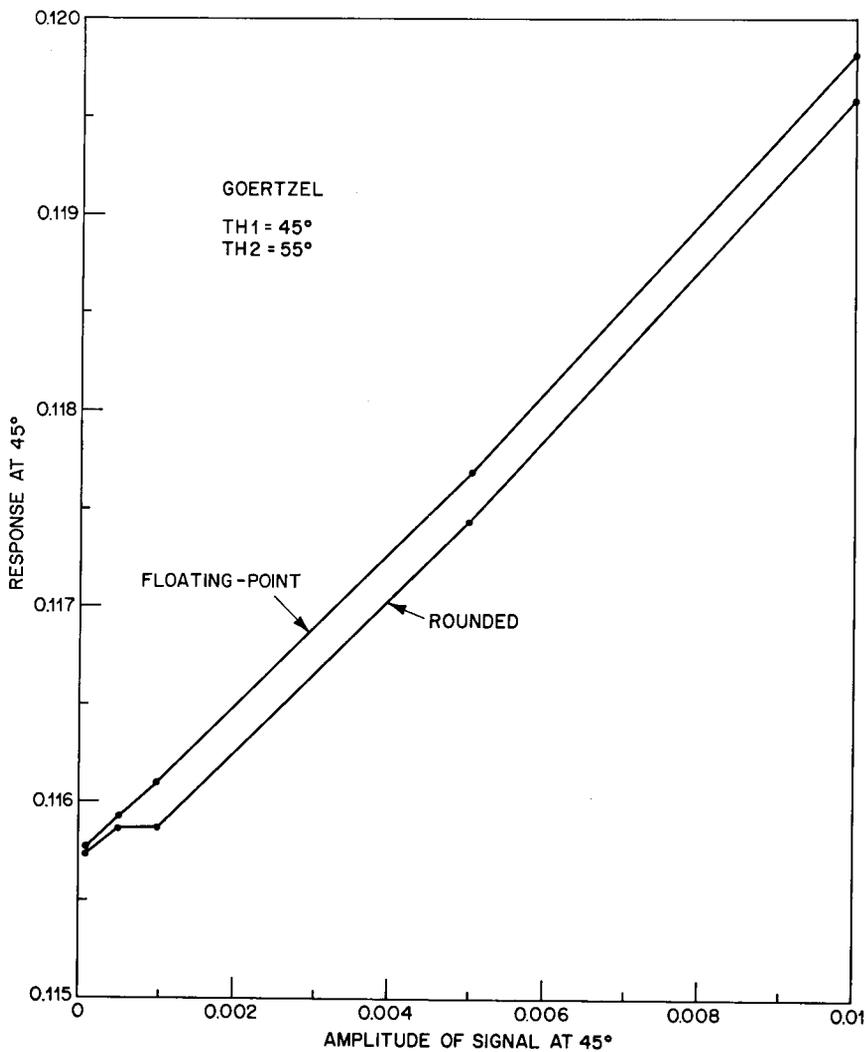


Fig. A5—Signal strength at 45°, the steering direction, vs magnitude of response with the interfering signal at 55° (magnified)

## Appendix B

### EXPERIMENTAL MEAN AND VARIANCE FOR A WEAK SIGNAL PLUS NOISE

The experimental mean and variance were calculated after 5000 trials for magnitudes varying from 0.0 to 0.1. Rounded arithmetic and automatic scaling were used for the recursive algorithm and the dot product, which were the only algorithms simulated. Tables B1-B4 give the results. Figures B1 through B8 depict the discrepancies between floating-point results and fixed-point simulator results.

Table B1

Amp. at 0°	Floating		Recursive		Dot Product	
	Mean	Variance	Mean	Variance	Mean	Variance
0.1	0.044373229 (0.11289062E-01)*	0.35699172E-06	0.044325061 (0.11199951E-06)	0.35620483E-06	0.044325122 (0.11199951E-01)	0.35621106E-06
0.03	0.034346217 (0.10160156E-02)	0.22412060E-06	0.034304321 (0.97656250E-03)	0.22358943E-06	0.034304332 (0.97656250E-05)	0.22359038E-06
0.01	0.033513388 (0.11289062E-03)	0.21300947E-06	0.033471893 (0.12207031E-03)	0.21249185E-06	0.033471893 (0.12207031E-03)	0.21249185E-06
0.003	0.033435261 (0.10160156E-04)	0.21194003E-06	0.033393378 (0)	0.21142062E-06	0.033393378 (0)	0.21142062E-06
0.001	0.033433259 (0.11289062E-05)	0.21190300E-06	0.033391370 (0)	0.21138431E-06	0.0033391370 (0)	0.21138431E-06
0.0003	0.033434692 (0.10160156E-06)	0.21191824E-06	0.033392950 (0)	0.21139953E-06	0.033392950 (0)	0.21139953E-06
0.0001	0.033435305 (0.11289062E-07)	0.21192527E-06	0.033393378 (0)	0.21140525E-06	0.033393378 (0)	0.21140525E-06
0	0.033435645 (0)	0.21192924E-06	0.033393555 (0)	0.21141028E-06	0.033393555 (0)	0.21141028E-06

\*Response without noise is given in parentheses.

Table B2

Amp. at 3°	Floating		Recursive		Dot Product	
	Mean	Variance	Mean	Variance	Mean	Variance
0.1	0.044421970 (0.011289062)*	0.36386636E-06	0.044363605 (0.11169434E-01)	0.36290567E-06	0.044372168 (0.11169434E-01)	0.36302644E-06
0.03	0.054018257 (0.10160156E-02)	0.23169706E-06	0.033968890 (0.97656250E-03)	0.23101597E-06	0.033975104 (0.10070801E-02)	0.23111941E-06
0.01	0.033077799 (0.11289062E-03)	0.22037314E-06	0.0333028516 (0.91552734E-04)	0.21970719E-06	0.033035657 (0.91552734E-04)	0.21980774E-06
0.003	0.032962001 (0.10160156E-04)	0.21918584E-06	0.032913635 (0)	0.21855469E-06	0.032920020 (0)	0.21862838E-06
0.001	0.032949237 (0.11289062E-05)	0.21911099E-06	0.032900769 (0)	0.21845687E-06	0.032906854 (0)	0.21853893E-06
0.0003	0.032946903 (0.10160156E-06)	0.21911256E-06	0.032898151 (0)	0.21848176E-06	0.032904663 (0)	0.21856441E-06
0.0001	0.032946439 (0.11289062E-07)	0.21911565E-06	0.032896600 (0)	0.21846975E-06	0.032903680 (0)	0.21856693E-06
0	0.032946241 (0)	0.21911764E-06	0.032896527 (0)	0.21846577E-06	0.032903131 (0)	0.21855820E-06

\*Response without noise is given in parentheses.

Table B3

Amp. at 15°	Floating		Recursive		Dot Product	
	Mean	Variance	Mean	Variance	Mean	Variance
0.1	0.044081811 (0.11289062E-01)*	0.35856037E-06	0.044016095 (0.11169434E-01)	0.35746658E-06	0.044031842 (0.11169434E-06)	0.35772813E-06
0.03	0.033611957 (0.10160156E-02)	0.21880179E-06	0.033558026 (0.97656250E-03)	0.21810768E-06	0.033571442 (0.97656250E-03)	0.21828128E-06
0.01	0.032652602 (0.11289062E-03)	0.20533061E-06	0.032598730 (0.91552734E-04)	0.20465385E-06	0.032611658 (0.91552734E-04)	0.20481564E-06
0.003	0.032530190 (0.10160156E-04)	0.20339398E-06	0.032475824 (0)	0.20269992E-06	0.032489081 (0)	0.20296734E-06
0.001	0.032515536 (0.11289062E-05)	0.20310526E-06	0.032461987 (0)	0.20242293E-06	0.032474451 (0)	0.20258414E-06
0.0003	0.032512541 (0.10160156E-06)	0.20303199E-06	0.032457568 (0)	0.20234545E-06	0.032472015 (0)	0.20251779E-06
0.0001	0.032511888 (0.11289062E-07)	0.20301370E-06	0.032457623 (0)	0.20232190E-06	0.032470575 (0)	0.20249740E-06
0	0.032511596 (0)	0.20300499E-06	0.032455701 (0)	0.20230259E-06	0.032469348 (0)	0.20248317E-06

\*Response without noise is given in parentheses.

Table B4

Amp. at 45°	Floating		Recursive		Dot Product	
	Mean	Variance	Mean	Variance	Mean	Variance
0.1	0.044266939 (0.11289062E-01)*	0.35720378E-06	0.044199896 (0.11169434E-01)	0.35611279E-06	0.044218701 (0.11169434E-01)	0.35638782E-06
0.03	0.034233958 (0.10160156E-02)	0.22546720E-06	0.034177771 (0.97656250E-03)	0.22473487E-06	0.034191229 (0.97656250E-03)	0.22490962E-06
0.01	0.033399423 (0.11289062E-03)	0.21425207E-06	0.033343774 (0.91552734E-04)	0.21354277E-06	0.033356311 (0.91552734E-04)	0.21370934E-06
0.003	0.033320699 (0.10160156E-04)	0.21310129E-06	0.033264722 (0)	0.21240511E-06	0.033278308 (0)	0.21255863E-06
0.001	0.033318527 (0.11289062E-05)	0.21303673E-06	0.033262708 (0)	0.21234886E-06	0.033275873 (0)	0.21249828E-06
0.0003	0.033319900 (0.10160156E-06)	0.21304188E-06	0.033264862 (0)	0.21234927E-06	0.033278003 (0)	0.21251219E-06
0.0001	0.033320495 (0.11289062E-07)	0.21304599E-06	0.033264868 (0)	0.21233795E-06	0.033277380 (0)	0.21250822E-06
0.0	0.033320827 (0)	0.21304849E-06	0.033264893 (0)	0.21233933E-06	0.033277698 (0)	0.21250711E-06

\*Response without noise is given in parentheses.

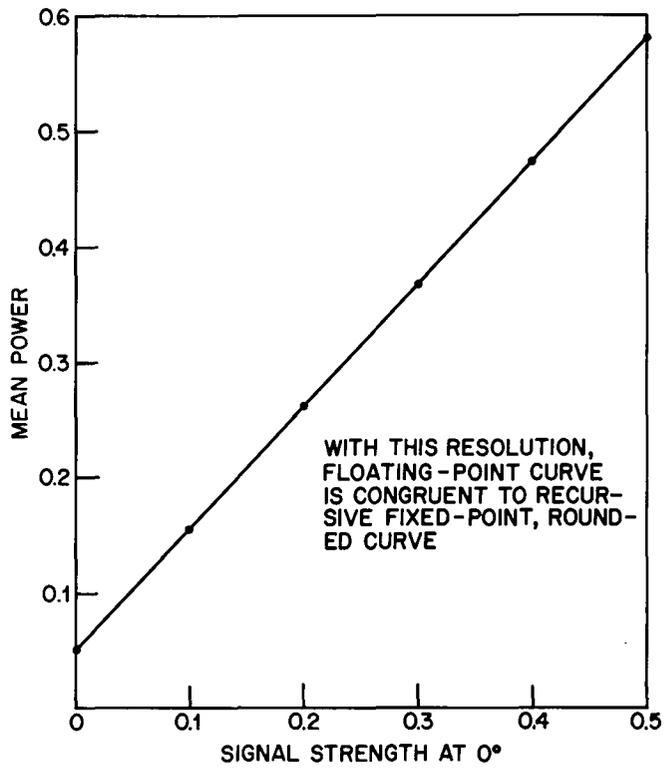


Fig. B1 — Signal strength at 0°, the steering direction, vs mean power.

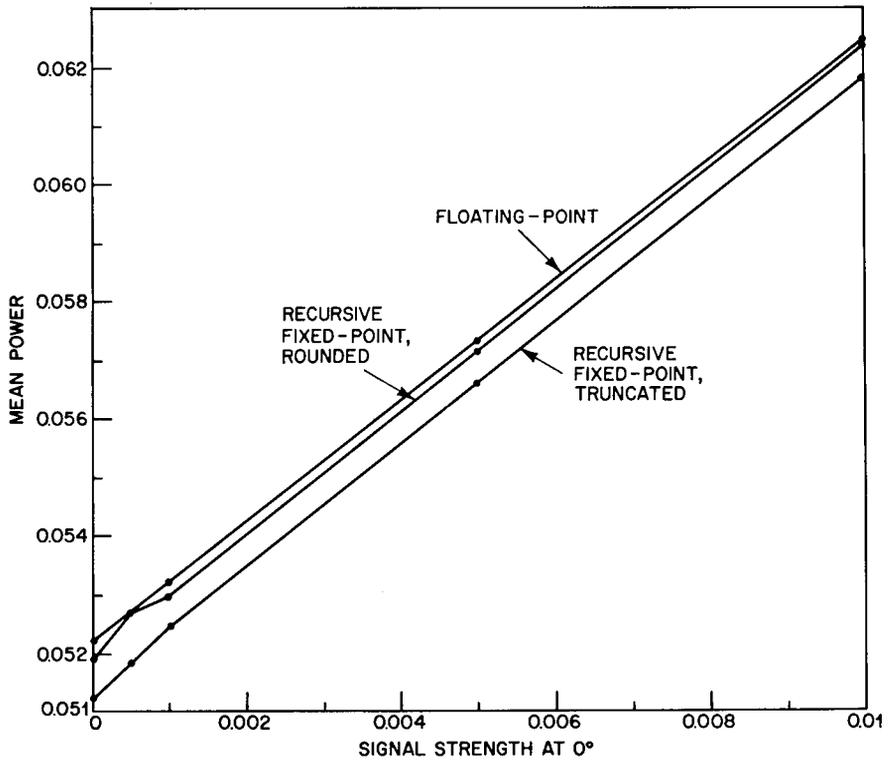


Fig. B2—Signal strength at 0°, the steering direction, vs mean power (magnified)

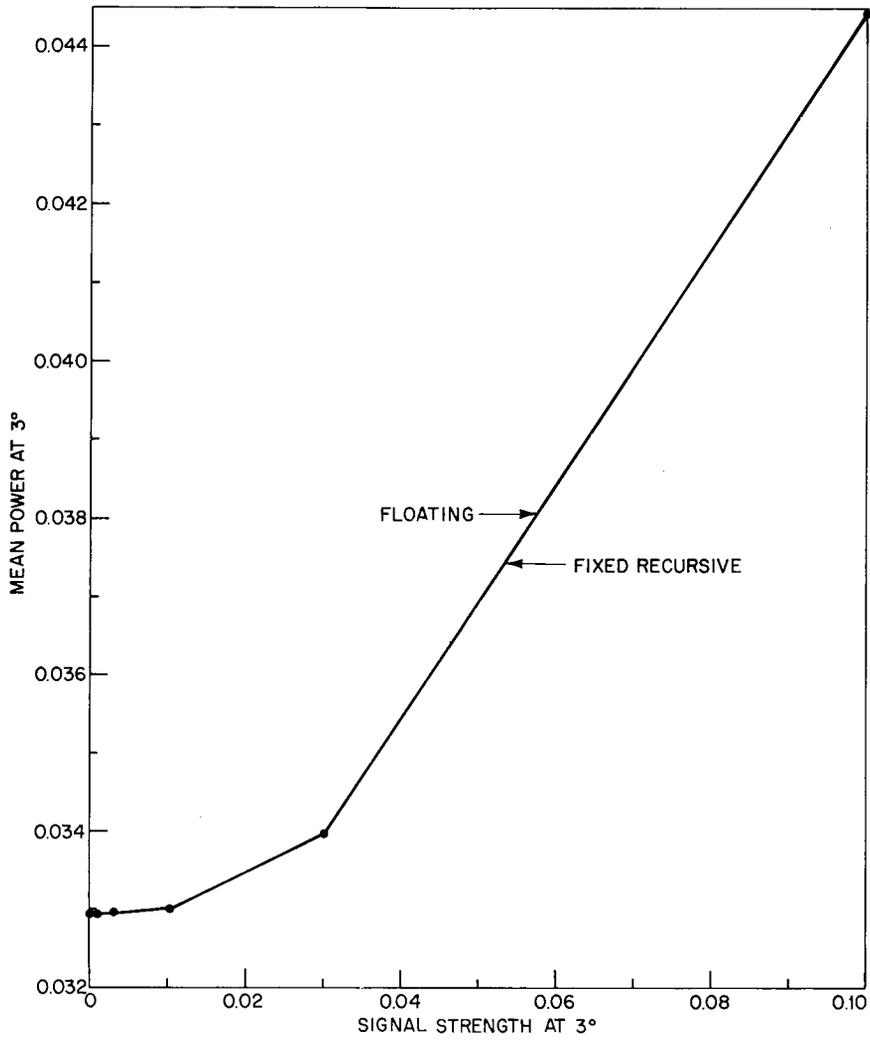


Fig. B3 — Signal strength at 3°, the steering direction, vs mean power.

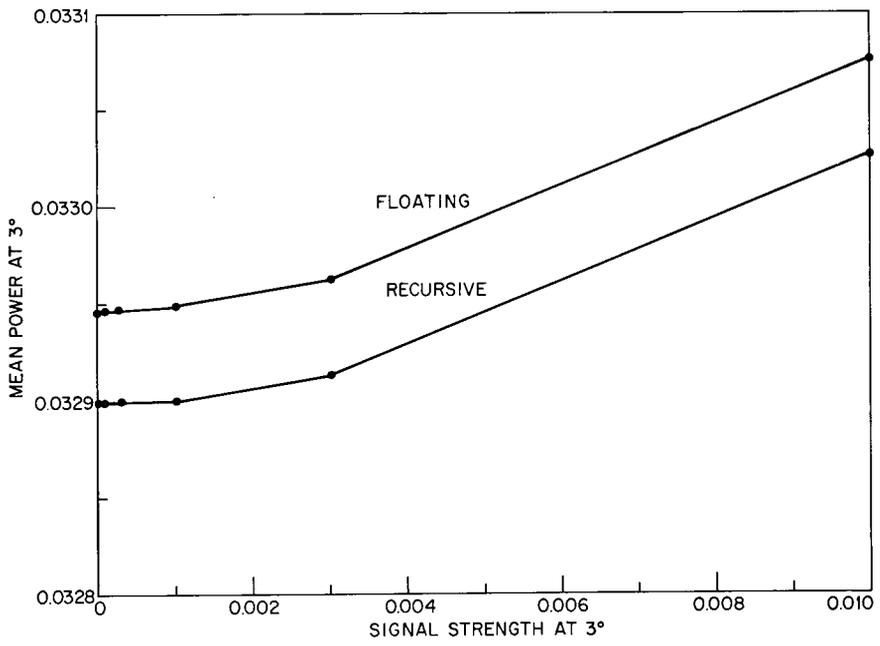


Fig. B4—Signal strength at 3°, the steering direction, vs mean power (magnified)

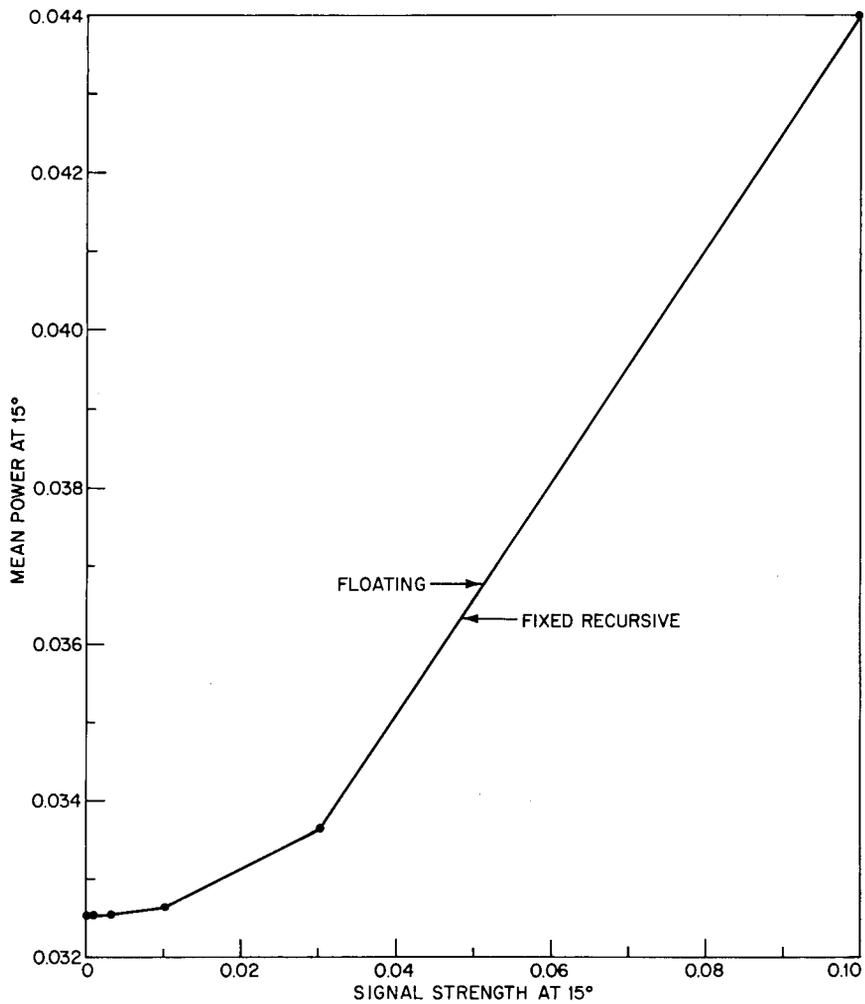


Fig. B5—Signal strength at 15°, the steering direction, vs mean power

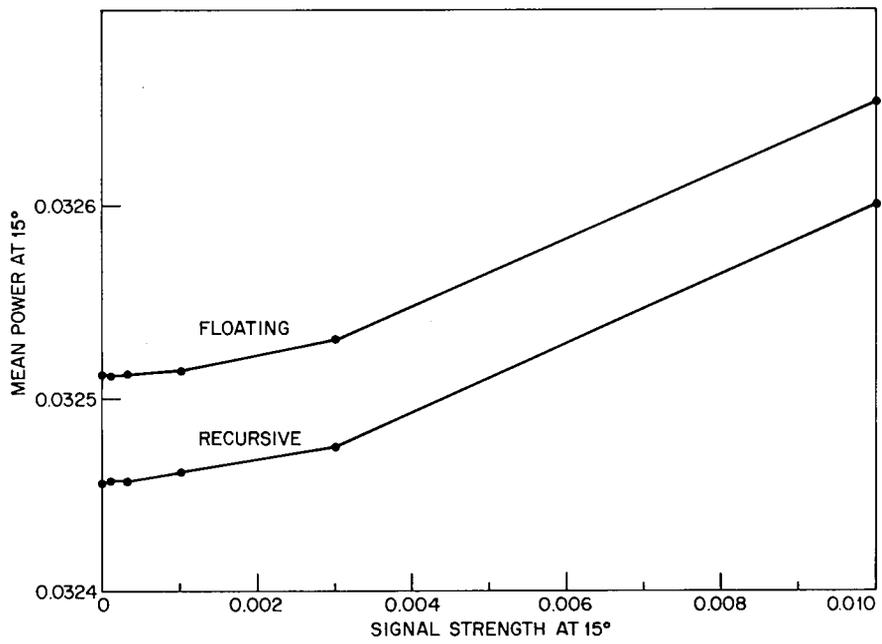


Fig. B6—Signal strength at 15°, the steering direction, vs mean power (magnified)

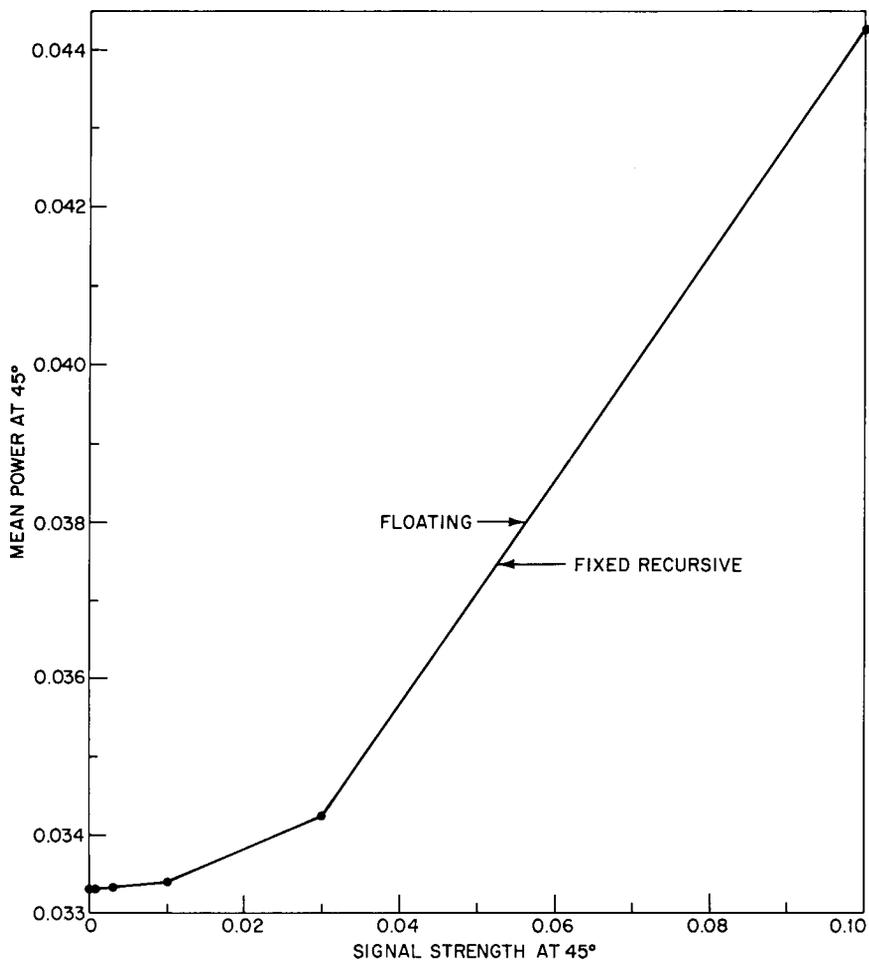


Fig. B7—Signal strength at 45°, the steering direction, vs mean power

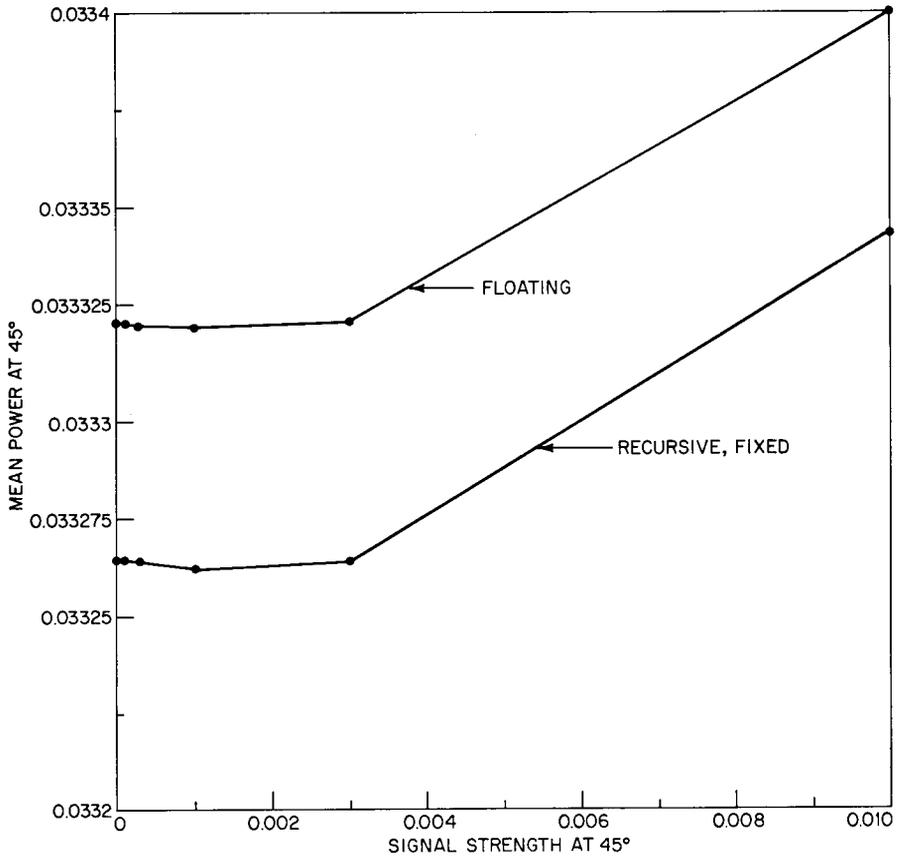


Fig. B8—Signal strength at 45°, the steering direction, vs mean power (magnified)