COMPUTER-AIDED TESTING OF A/D CONVERTERS USING SINUSOIDAL FITTING

by

Michael L. Handler and Michael J. Sabin

Memorandum No. UCB/ERL M85/68

16 August 1985

COMPUTER-AIDED TESTING OF A/D CONVERTERS USING SINUSOIDAL FITTING

by

Michael L. Handler and Michael J. Sabin

Memorandum No. UCB/ERL M85/68

16 August 1985

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Computer-Aided Testing of A/D Converters Using Sinusoidal Fitting

Michael L. Handler and Michael J. Sabin

Department of Electrical Engineering and Computer Sciences

and the

Electronics Research Laboratory

University of California, Berkeley 94720

Abstract

A useful indicator of the dynamic performance of an analog-to-digital converter (ADC) is the degree to which it distorts a sinusoid. This paper describes a computer program which accurately computes the minimum mean square error between the converter's response to a sinusoid and the sinusoid that best fits this response. Such a performance test is easy to conduct in that the only data the program requires are the ADC output samples. Moreover, because a large number of samples is used (several hundred thousand), the program's results are very accurate. Accuracy is verified by comparing the program's results for a simulated ideal ADC to the theoretical mean square error and by testing an actual converter.

This work was supported by the Semiconductor Research Corporation

1. Introduction

This paper describes a computer program for accurately measuring the total harmonic distortion of an analog-to-digital converter (ADC). A spectrally pure sine wave is input to the ADC under test. The sample rate of the ADC is set greater than the Nyquist rate but is not synchronized to the input signal. Using least-squares minimization, the computer program fits a discrete-time sinusoid to the ADC output sequence. Letting $\left\{ f(n) \right\}_{n=0}^{N-1}$ denote the ADC output sequence and $\left\{ C + G\cos(\omega n + \Phi) \right\}_{n=0}^{N-1}$ denote the fitted sinusoid, the parameters G, ω, Φ, and C are selected by the program to minimize

$$\frac{1}{N}\sum_{n=0}^{N-1}\left\{ f(n) - \left[C + G\cos(\omega n + \Phi) \right] \right\}^2 . \tag{1.1}$$

The program is designed to accommodate large values of N, typically in the range of 500,000.

The minimum mean square error (MMSE) that results when (1.1) is minimized indicates the degree to which the ADC distorts the sine wave input. Ideally, the MMSE should be a function of only the number of bits of the converter's resolution. Practically, an ADC introduces errors which contribute to an increase in the MMSE: nonlinearity in the overall transfer characteristic (integral nonlinearity); uneven spacing of the quantization thresholds (differential nonlinearity); output values which never occur (missing codes); and inconsistency in the instant at which the input is quantized (aperture jitter) [2],[3],[8]. These various dynamic errors all contribute to a degradation in an ADC's performance, and their cumulative effects are reflected in an increase of the MMSE above its ideal value. The relative importance of each of these types of errors depends on the application (see, e.g., [8]). Recent advances in monitoring the errors individually can be found in [1].

2. Testing Procedure

The testing procedure consists of driving the ADC with a sinusoid meeting the requirements described below and capturing the converter's output sequence in a file. The data analysis program reads this file and then applies the iterative sinusoidal-fitting algorithm also described below. Each iteration decreases the value of the mean square error (MSE) until convergence to the MMSE occurs. Convergence is judged to have occurred when numerical noise causes an apparent increase in the MSE--a criterion easily changed depending on the application. The only parameters which the program needs as input are the file name of the samples and the number of bits of the ADC. If necessary, the input file must be converted to two's complement format prior to being processed by the program.

As previously mentioned, the input sine wave should be spectrally pure. In particular, the ratio of the energy in the fundamental frequency to the energy in the largest amplitude harmonic must exceed 6.02B dB ($= 20 \log 2^B$), where B is the number of bits in the ADC. This restriction insures that impurity of the sine wave is negligible within the ADC's resolution [2]. Ultra-low distortion oscillators meeting this requirement for values of B up to 16 are available commercially [1].

The sine-wave amplitude should be as large as the converter can accept without overloading so as to encompass the complete decision range of the device under test. Full-scale testing also produces the maximum slew rate for a given input frequency and sample rate. It is not necessary that the input have no dc offset since the fitted sinusoid includes a dc component. However, the presence of an offset requires a reduction in the ac amplitude and thus does not test the full range of the ADC. When using ADCs with offset binary or two's complement coding, in which the output range is asymmetrical about the mid-scale value, the input analog sine wave must be adjusted to eliminate ADC outputs of negative full scale (e.g., binary 1000 for four-bit two's complement). The histogram test [1], which compiles a cumulative tally of occurrences of each output code, would be convenient in adjusting the amplitude.

Under various sample rates and analog frequencies, the converter sees different digital frequencies, expressed in cycles/sample. For a given amplitude, a greater digital frequency will result in a faster slew rate to which the ADC must respond. A converter's errors will differ at various slew rates. The digital frequency should thus be set near the value of interest, and the resulting MMSE should be interpreted for that digital frequency. Of course, for an ideal ADC, the MMSE would not depend on frequency.

It is important that the sample rate, f_s, and the analog sine wave frequency, f, be nonharmonically related. For the gathering of N samples, this requires

$$\frac{f}{f_s} n \neq an\ integer, \qquad 0 < n \leq N-1. \tag{2.1}$$

Such a requirement will minimize the repetition of ADC output codes. It is important to avoid a repetition of specific codes because the converter's errors associated with these codes would have too great an influence on the MMSE. Also, care must be taken to insure that the phase jitter in the analog input is negligible.

It is tempting to use a statistical approach to determine the number of samples needed to accurately characterize an ADC's MMSE. Unfortunately, such an approach seems unrealistically optimistic. To see this, assume that the squared-error sequence in (1.1) is a sequence of random variables $\{Y_n\}$ which are independent and identically distributed (i.i.d.). We want to pick N large enough so that the variance of $S_N \triangleq N^{-1} \sum_{n=0}^{N-1} Y_n$ is small; this will ensure that the average energy in the observed error is typical of what is produced by the ADC. The variance of S_N is $N^{-1}\sigma^2$, where σ^2 is the variance of Y_n [6]. The N^{-1} factor implies that this variance is negligible when using only several tens of samples--a number much smaller than the number of ADC output codes. For a practical ADC, the i.i.d. assumption seems invalid, since the nature of each error will depend on the output code. Thus, it seems more appropriate to require that the total number of samples be several times the number of possible output codes. This will ensure that the error associated with each output code contributes to the estimated MMSE.

3. The Algorithm

The problem is to find G, C, ω, and Φ to minimize (1.1). The approach is simplified by expressing the sinusoid in terms of its quadrature components; that is, find A, B, C, and ω to minimize

$$E(\omega, A, B, C) \triangleq \frac{1}{N}\sum_{n=0}^{N-1}\left\{f(n) - \left[C + A\cos(\omega n) + B\sin(\omega n)\right]\right\}^2, \qquad (3.1)$$

where

$$\Phi = -\tan^{-1}\left(\frac{B}{A}\right) \quad \text{and} \quad G = \sqrt{A^2 + B^2}.$$

The strategy of the algorithm is to iteratively search for the value of ω that minimizes the function ξ defined by

$$\xi(\omega) = \min_{A,B,C} E(\omega, A, B, C). \qquad (3.2)$$

Once the value ω^* that minimizes ξ is found, then the accompanying values A^*, B^*, and C^* that achieve the minimum in (3.2) can be used to construct the fitted sinusoid; $\xi(\omega^*)$ is the MMSE. Fig. 1 provides a flowchart of the algorithm. We now present a detailed description of each block in the flowchart.

Block 1: By converting the ADC output sequence to a square wave and measuring its frequency, the program determines an initial frequency guess which is extremely close to the true frequency. For the conversion, the software emulates a Schmitt trigger in that an upper threshold must be exceeded to enter the high state of the square wave and a lower threshold must be negatively exceeded to enter the low state. This approach is superior to using zero crossings because the hysteresis of the Schmitt trigger provides immunity to noise and ADC errors. The software tallies the number of complete square-wave cycles and then divides by the M samples within that integral number of cycles. This initial frequency guess is multiplied by 2π to yield ω_1, in radians per sample.

Block 2: The procedure used to search for ω^* requires three guesses which are very close. To this end, the program selects $\omega_0 = \omega_1(1-\frac{1}{M})$ and $\omega_2 = \omega_1(1+\frac{1}{M})$; ω_0, ω_1, and ω_2 will be the initial guesses. The choice of ω_0 and ω_2 is motivated by the observation that the measurement of ω_1 is accurate to a resolution of $\frac{\omega_1}{M}$.

Block 3: For each ω_i candidate, $i=0,1,2$, $\xi(\omega_i)$ is found. See Section 3.1 below.

Block 4: Using the quadratic-fit method, the program computes ω_3, the next guess of ω^* based on ω_0, ω_1, and ω_2. See Section 3.2 below.

Block 5: For the new ω_3, $\xi(\omega_3)$ is found.

Block 6: The program now has four pairs of $(\omega, \xi(\omega))$. The pair with the largest $\xi(\omega)$ is discarded.

Block 7: In order to perform the quadratic fit on the remaining three pairs of $(\omega, \xi(\omega))$, the program re-indexes them according to increasing ω.

3.1. Computation of $\xi(\omega)$

For ω fixed, the problem is to find values for A, B, and C which achieve the minimum in (3.2). Minimizing (3.1) term by term allows the problem to be formulated as: find A, B, and C such that

$$\begin{bmatrix} \cos 0\omega & \sin 0\omega & 1 \\ \cos 1\omega & \sin 1\omega & 1 \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cos(N-1)\omega & \sin(N-1)\omega & 1 \end{bmatrix} \cdot \begin{bmatrix} A \\ B \\ C \end{bmatrix} \approx \begin{bmatrix} f(0) \\ f(1) \\ \cdot \\ \cdot \\ f(N-1) \end{bmatrix}, \tag{3.3}$$

or in matrix symbols,

$$\mathbf{M}\,\mathbf{x} \approx \mathbf{b}. \tag{3.4}$$

Because of the inherent quantization in the $f(n)$'s, equality in (3.3) is impossible. Equation (3.3) implies that minimizing (3.1) at a fixed ω is a projection problem, with solution $\mathbf{x}^*$ given by [9]

$$\mathbf{M}^T\mathbf{M}\mathbf{x}^* = \mathbf{M}^T\mathbf{b}. \tag{3.5}$$

Reintroducing the factor $\frac{1}{N}$ on both sides of (3.5) and carrying out the required matrix multiplications, (3.5) becomes

$$\begin{bmatrix} \gamma_{00} & \gamma_{01} & \gamma_{02} \\ \gamma_{10} & \gamma_{11} & \gamma_{12} \\ \gamma_{20} & \gamma_{21} & \gamma_{22} \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} <f(n)\cos(n\omega)> \\ <f(n)\sin(n\omega)> \\ <f(n)> \end{bmatrix}, \tag{3.6}$$

where:

$$<\zeta(n)> \triangleq \frac{1}{N}\sum_{n=0}^{N-1}\zeta(n),$$

$$\gamma_{00} = <\cos^2(n\omega)> = \frac{1}{2} + \frac{1}{2}\frac{\sin(\omega N)\cos(\omega(N-1))}{N\sin(\omega)},$$

$$\gamma_{11} = <\sin^2(n\omega)> = \frac{1}{2} - \frac{1}{2}\frac{\sin(\omega N)\cos(\omega(N-1))}{N\sin(\omega)},$$

$$\gamma_{22} = 1,$$

$$\gamma_{10} = \gamma_{01} = <\cos(n\omega)\sin(n\omega)> = \frac{1}{2}\frac{\sin(\omega N)\sin(\omega(N-1))}{N\sin(\omega)},$$

$$\gamma_{20} = \gamma_{02} = <\cos(n\omega)> = \frac{\sin(\omega N/2)\cos(\omega(N-1)/2)}{N\sin(\omega/2)},$$

$$\gamma_{21} = \gamma_{12} = <\sin(n\omega)> = \frac{\sin(\omega N/2)\sin(\omega(N-1)/2)}{N\sin(\omega/2)}.$$

The program explicitly computes the averages appearing on the right-hand side of (3.6) and uses the closed-form expressions listed above for computing the entries in the square matrix. (The closed-form expressions are from [7].) Then, the program uses Cramer's rule to solve for A, B, and C.

A benefit to treating the above problem as a projection is the resulting ease in computing $\xi(\omega)$ once A, B, and C are found. The squared error between the best-fit sine wave and the ADC sequence is just, from the Pythagorean theorem,

$$||\mathbf{M}\mathbf{x}^* - \mathbf{b}||^2 = ||\mathbf{b}||^2 - ||\mathbf{M}\mathbf{x}^*||^2. \tag{3.7}$$

$\xi(\omega)$ is

$$\begin{aligned}\frac{1}{N}||\mathbf{M}\mathbf{x}^* - \mathbf{b}||^2 &= <\mathbf{b}^2> - <(\mathbf{M}\mathbf{x}^*)^{\mathrm{T}}(\mathbf{M}\mathbf{x}^*)> \\ &= <\mathbf{b}^2> - <\mathbf{x}^{*\mathrm{T}}\mathbf{M}^{\mathrm{T}}\mathbf{M}\mathbf{x}^*> \\ &= <\mathbf{b}^2> - <\mathbf{x}^{*\mathrm{T}}\mathbf{M}^{\mathrm{T}}\mathbf{b}>.\end{aligned} \tag{3.8}$$

The matrix multiplications in (3.8) lead to

$$\xi(\omega) = <f^2(n)> - A<f(n)\cos(n\omega)> - B<f(n)\sin(n\omega)> - C<f(n)>. \tag{3.9}$$

Note that the terms $<f^2(n)>$ and $<f(n)>$ need only be computed once, since they do not change with ω. Also, the terms $<f(n)\cos(n\omega)>$ and $<f(n)\sin(n\omega)>$ were already computed for the calculations of A, B, and C, as can be seen in (3.6). Therefore, once A, B, and C are computed, (3.9) provides a much quicker and more convenient approach to determining $\xi(\omega)$ than does (3.1).

3.2. Quadratic Fit

Fig. 2 shows typical behavior of $\xi(\omega)$ in the vicinity of ω^*. The parabolic shape motivates the use of the quadratic-fit method [4] to zero in on ω^*. This method applies to the problem at hand as follows. Given three coordinate pairs of $(\omega, \xi(\omega))$, a parabola of the form $\xi(\omega) = a\omega^2 + b\omega + c$ can be uniquely determined to fit these points. The parameters a, b, and c are determined by solving the following system of equations:

$$\begin{bmatrix} \omega_0^2 & \omega_0 & 1 \\ \omega_1^2 & \omega_1 & 1 \\ \omega_2^2 & \omega_2 & 1 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \xi(\omega_0) \\ \xi(\omega_1) \\ \xi(\omega_2) \end{bmatrix}. \tag{3.10}$$

The minimum of the parabola occurs at $\omega_3 = \frac{-b}{2a}$. Solving (3.10) for a and b and substituting into the expression for ω_3 yields

$$\omega_3 = \omega_1 + \frac{1}{2}\left[\frac{x_2^2 y_1 - x_1^2 y_2}{x_2 y_1 - x_1 y_2}\right], \tag{3.11}$$

where:

$$x_1 = \omega_0 - \omega_1,$$

$$x_2 = \omega_2 - \omega_1,$$

$$y_1 = \xi(\omega_1) - \xi(\omega_0),$$

$$y_2 = \xi(\omega_1) - \xi(\omega_2).$$

The program implements the quadratic fit by using (3.11). Prior to a repetition of the quadratic fit in the course of the algorithm, the pair $(\omega, \xi(\omega))$ with the largest $\xi(\omega)$ is discarded (Block 6 in Fig. 1). The three remaining sets are then used for the next fit.

3.3. Numerical Considerations

Accuracy in the calculations is the paramount goal in the programming effort. One obvious approach to maximizing accuracy is to perform all calculations with the greatest precision possible. The sine-fitting program, written in C, uses double precision for all computations. On the VAX 11/750, this resulted in 64 bits of precision. With 64 bits of precision, the best-fit digital sine wave is indeed accurate and, for all practical purposes, unquantized.

The calculations of the average terms $<f(n)\cos(n\omega)>$ and $<f(n)\sin(n\omega)>$ in (3.6) illustrate the need for special precautions. Each of these averages requires the summation of tens or hundreds of thousands of terms. Without some care, roundoff error will be troublesome. The approach used here is to break the summation into many summations, one summation for each of the different values of $f(n)$. That is, denoting the smallest $f(n)$ value as $f_{\min}$ and the largest $f(n)$ value as $f_{\max}$, $<f(n)\cos(n\omega)>$ is calculated as

$$\sum_{k=f_{\min}}^{f_{\max}} k\left[\sum_{n:f(n)=k} \cos(\omega n)\right]. \tag{3.12}$$

The program maintains an array, each element of the array corresponding to a k, so that the bracketed terms may be computed first. The same technique is used to compute $<f(n)\sin(n\omega)>$.

Furthermore, because the calculations of $<f(n)\cos(n\omega)>$ and $<f(n)\sin(n\omega)>$ account for most of the CPU time, programming effort concentrated on accelerating these computations. Experimental observation led to the conclusion that a cosine or sine call to the computer system's math library requires about fifteen times longer to complete (in CPU time) than does a double precision multiply. Fortunately, the fact that ω remains constant and that n increases at a constant rate in the above averages allows for a recursive calculation of the cosine and sine using multiplies. The desired relation uses a complex sinusoid:

$$e^{j\omega n} = Ke^{j\omega(n-1)}, \tag{3.13}$$

where K is the complex exponential $e^{j\omega}$. Not only does (3.13) eliminate the need for system sine and cosine calls, but it also provides both the cosine and sine after just one complex multiply, which amounts to four real multiplications and two real additions. However, accumulated errors in the recursion cause the computed value of $e^{j\omega n}$ to wander from its correct value. It is therefore necessary to occasionally adjust the value of $e^{j\omega n}$ by explicitly computing it via the system library.

A short test program was used to find an appropriate value for the maximum number of repeated recursions of (3.13) which should be allowed. With $\omega = 0.3$ (a typical value), the value of $|e^{j\omega n}|$ was compared to 1 for increasing values of n. It was found that for $n \leq 80$, $|e^{j\omega n}|$ differed from 1 by less than 10^{-14}. This accuracy is comparable to that of the double-precision representation, and so it was decided that eighty repeated recursions of (3.13) should be allowed.

When using the above approach to computing sines and cosines quickly, the sine-fitting program required approximately eight seconds of real time for each iteration of the algorithm with N=100,000. The time requirement is roughly proportional to the number of samples. Without using (3.13), the program requires about six times longer to complete.

As a final note, the quadratic fit formulas for (3.11) depend on $\xi(\omega)$ only through the difference terms $\xi(\omega_1) - \xi(\omega_0)$ and $\xi(\omega_1) - \xi(\omega_2)$. Equation (3.9) shows that the quantity $<f^2(n)>$ cancels when these difference terms are computed. This is exploited by storing the value

$$\psi(\omega) = A<f(n)\cos(n\omega)> + B<f(n)\sin(n\omega)> + C<f(n)>, \qquad (3.14)$$

which is computed during the process of calculating $\xi(\omega)$. Then, $\xi(\omega_i) - \xi(\omega_j)$ is computed as $\psi(\omega_j) - \psi(\omega_i)$, a numerically superior expression.

3.4. Relation to DFT

The discrete Fourier transform (DFT) of $\{f(n)\}_{n=0}^{N-1}$ is obtained by solving (3.6) for A and B at values of ω equal to $\frac{2\pi m}{N}$, where m is an integer. To see this, observe that for these values of ω, the off-diagonal terms in the square matrix are zero. This yields

$$A = 2<f(n)\cos(\frac{2\pi m}{N}n)>, \tag{3.15}$$

$$B = 2<f(n)\sin(\frac{2\pi m}{N}n)>,$$

which are recognized as the quadrature amplitudes of the mth DFT component (except for a constant factor). Practically, it is no easier to solve (3.6) for this special choice of ω because, regardless of the value of ω, virtually all of the computational effort is in computing the right-hand side of (3.6); computing the off-diagonal terms and applying Cramer's rule is negligible in comparison. The conclusion is that the effort in computing $\xi(\omega)$ for any choice of ω is the same as that in computing a single DFT point.

Although the DFT can be computed by using the fast Fourier transform (FFT), the FFT is efficient only when computing many DFT points. The sine-fit algorithm typically requires determining only five values of $\xi(\omega)$, which is computationally as difficult as determining five DFT points. Performing a large FFT is much more computationally intensive than computing a few DFT points. Hence, an FFT-based scheme would require much more computation than the sine-fit program.

In an actual testing situation, the sine-wave-generator frequency and the sampling frequency are not synchronized, and the digital frequency of the sinusoid will not be a DFT frequency (i.e., will not be $\frac{2\pi m}{N}$ for any integer m). This creates a problem if a DFT is used to estimate the amplitude of the fundamental. To illustrate, Fig. 3 shows the magnitude spectrum resulting from 32K samples from an ideal 8-bit converter. The magnitudes were calculated by using (3.15). The input to the ideal ADC was $127\cos(0.0501n)$. However, Fig. 3 indicates that the peak magnitude is only 100.5. In contrast, when the sine-fitting program used the same ADC data, the result was a peak amplitude of 127.0. Thus, the peak amplitude of the DFT is not a good estimate of the amplitude of the fundamental, whereas the amplitude of the fitted sinusoid is a good estimate.

4. Simulation Results

Simulating an ideal B-bit ADC allows for comparing the program's report of the MMSE to the theoretical value.

4.1. The Simulated ADC

In a B-bit ADC, there are 2^B-1 thresholds, or transition points, for the analog input value. To allow for a zero-valued analog input to correspond to a digital output of zero (in two's complement), the simulation package sets the quantizer characteristics to the "mid-tread" class. Unfortunately, one more negative level than positive level thus appears, as Fig. 4 indicates for an ideal 3-bit ADC. The simulation program just ignores the possibility of this extra negative level in order to maintain symmetry for sinusoidal inputs. Therefore, the program sets the number of thresholds to 2^B-2; the most negative level is at $-(2^{(B-1)}-2+0.5)$ and the most positive level is at $+(2^{(B-1)}-2+0.5)$. One least significant bit (1 LSB), which represents the quantization step size, is one unit in the simulation. Finally, there are 2^B-1 possible output codes because of the discarded quantization level.

Although the gain and offset of the simulated perfect analog sine wave may be manually set, the program's default values result in no offset and a gain equal to full scale. Here, full scale corresponds to the analog input equal in value to the maximum output, which is $2^{(B-1)}-1$ units. Fig. 5 shows the sine-fitting program's output for an ideal 12-bit converter with input $2047\cos(0.24821044420n)$. Note that the MMSE practically converges after just the first repetition of the algorithm.

4.2. The Theoretical MSE

The theoretical value for the MSE of the above ideal ADC with full-scale input is derived as follows, where it is clear that the MSE here corresponds to the program's MMSE. The sampled, unquantized input to the ADC is considered a random variable **X** which is defined by

$$\mathbf{X} = G\cos\Theta, \tag{4.1}$$

where G is the full-scale amplitude and Θ is a random variable uniformly distributed between 0 and 2π. This model is a good approximation to the analog input only if the frequency is asynchronous to the sample rate and if the total number of samples span a large number of cycles. The more samples, the better the approximation.

The transformation from Θ to **X** leads to the probability density function of **X** [5]:

$$f_{\mathbf{X}}(x) = \begin{cases} \dfrac{1}{\pi\sqrt{G^2-x^2}}, & |x| \le G \\ 0, & |x| > G \end{cases} \tag{4.2}$$

The mean square error for the full-scale, zero-average signal in (4.1) can be now be expressed as

$$\mathrm{MSE} = \sum_{k=-G}^{G} \int_{k-\frac{1}{2}}^{k+\frac{1}{2}} (x-k)^2 f_{\mathbf{X}}(x)\,dx \tag{4.3}$$

$$= \frac{1}{\pi} \sum_{k=-G}^{G} \left[\sqrt{G^2-x^2}\left(\frac{-x}{2}+2k\right) + \left(\frac{G^2}{2}+k^2\right)\sin^{-1}\left(\frac{x}{G}\right) \right]_{x=k-\frac{1}{2}}^{x=k+\frac{1}{2}} . \qquad (4.4)$$

Table 1 lists the theoretical MSE's calculated from (4.4) for various values of B.

As B increases, the quantization becomes finer, allowing the quantization error to be modeled as having a uniform probability density function within each band. With such an approximation, the energy in the quantization error for a uniform quantizer having threshold step size Δ is the familiar expression $\frac{\Delta}{12}$. For the simulation, Δ is one unit, resulting in a quantization error energy of $\frac{1}{12}$. This is exactly the value the MSE is approaching in Table 1 for large B.

Also listed in Table 1 are the actual results of the sine-fitting program for an ideal ADC under an ideal input. The results indicate that the algorithm strongly agrees with the theoretical expectations. The slight differences between the theoretical and simulated MSE's arise because of the finite number of samples.

5. Example On Practical ADC

For the purpose of verifying the sine-fitting program's performance using actual ADC data, we used an acquisition system consisting of an LSI-11 recording the output of a 12-bit, bipolar, laser-trimmed, R-2R converter. This system limits the sample rate to less than 40 KHz and limits the number of samples to 64K. (In the future, a system allowing sample rates approaching 1 MHz and allowing nearly 1000K samples will be used.)

Table 2 lists the MMSE for two test runs at various digital frequencies. The data shows that this converter's performance is better at smaller digital frequencies than at larger ones. The MMSE can also be used to indicate the ADC's effective number of bits, which is approximately given by

$$\textit{effective bits} \approx B - \frac{1}{2}\log_2(12 \cdot \text{MMSE}), \tag{5.1}$$

where B is the actual number of bits. This equation is based on the approximation of a uniform probability density function of the quantization error within each band, as discussed above. At the higher frequencies, the above ADC's effective number of bits is below 11 bits while at the lower frequencies, its effective number of bits exceeds 11.

Errors in the digital sampling hardware in the test circuitry could have catastrophic effects on the MMSE. For example, when the LSI-controlled test system was used at a low sampling rate of 4.883 KHz, the data consistently resulted in an MMSE on the order of 10,000 to 100,000. It turned out that the sampling hardware erred at these low sampling rates by causing samples to be occasionally recorded twice. In fact, for one file of 10,000 samples at a digital period of 49.20 samples/cycle, the MMSE was 39,080.12. However, after ten samples which were clearly duplicates were removed, the MMSE was then 0.2890, which is consistent with the entries in Table 2. It is worth noting that the histogram test never detected this hardware error, since it is insensitive to such an anomaly. Aperture jitter is a similar error to which the sine-fitting program is sensitive and the histogram test is not.

6. Summary

This paper presented an algorithm to compute the minimum mean square error of an ADC's response to a sinusoid. An efficient and accurate sine-fitting computer program employing this algorithm was also presented. In addition to accuracy in the computations, the large number of samples used contributes to greater accuracy in the description of the converter's performance. The accuracy of the program and algorithm was verified by using both simulated and experimental data.

The minimum mean square error is a useful indicator of the harmonic distortion caused by an ADC. An effective number of bits can be calculated using this value. Errors caused by the ADC manifest themselves as an increase in the MMSE from its theoretical value.

The sine-fitting program provides a more accurate description of an ADC's total harmonic distortion than does a DFT test. Furthermore, the sine-fitting program is computationally more efficient for determining the fundamental amplitude than a DFT implemented using the FFT. Errors in the digital sampling hardware and aperture errors have a significant impact on the MMSE; the histogram test is insensitive to these errors.

REFERENCES

[1] Doernberg, J., H-S Lee, and D. A. Hodges, "Full-Speed Testing of A/D Converters," *IEEE Transactions on Solid-State Circuits*, Vol. SC-19, No. 6, December 1984, pp. 820-829.

[2] *Dynamic Performance Testing of A to D Converters*, Hewlett Packard Product Note 5180A-2.

[3] Kester, Walter, "Test Video A/D Converters Under Dynamic Conditions," *Electronic Design News*, August 18, 1982.

[4] Luenberger, David G., *Introduction To Linear and Nonlinear Programming*, Addison-Wesley Publishing Co., Inc., 1973.

[5] Papoulis, A. S., *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, Inc., 1974, p. 133.

[6] Ross, Sheldon, *A First Course in Probability*, Macmillan Publishing Co., Inc., 1976, p. 236.

[7] Ryzhik, G., *Table of Integrals, Series, and Products*, Academic Press, Inc., New York, 1965.

[8] Sheingold, Daniel H., editor, *Analog-Digital Conversion Notes*, Analog Devices, Inc., Norwood, Mass., 1980.

[9] Strang, Gilbert, *Linear Algebra and Its Applications*, Academic Press, Inc., New York, 1980, p. 114.

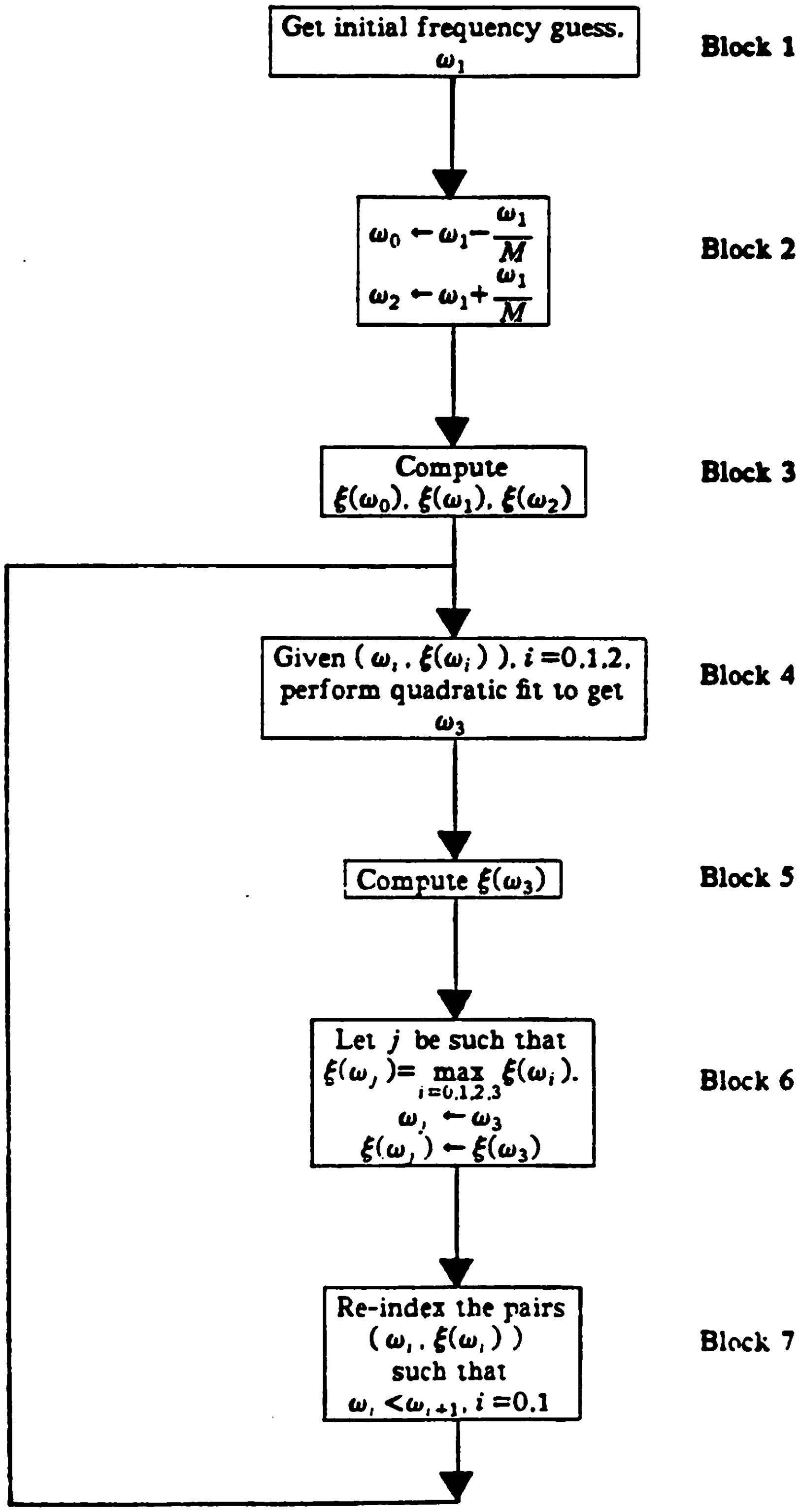

Fig. 1. Detailed flowchart of algorithm.

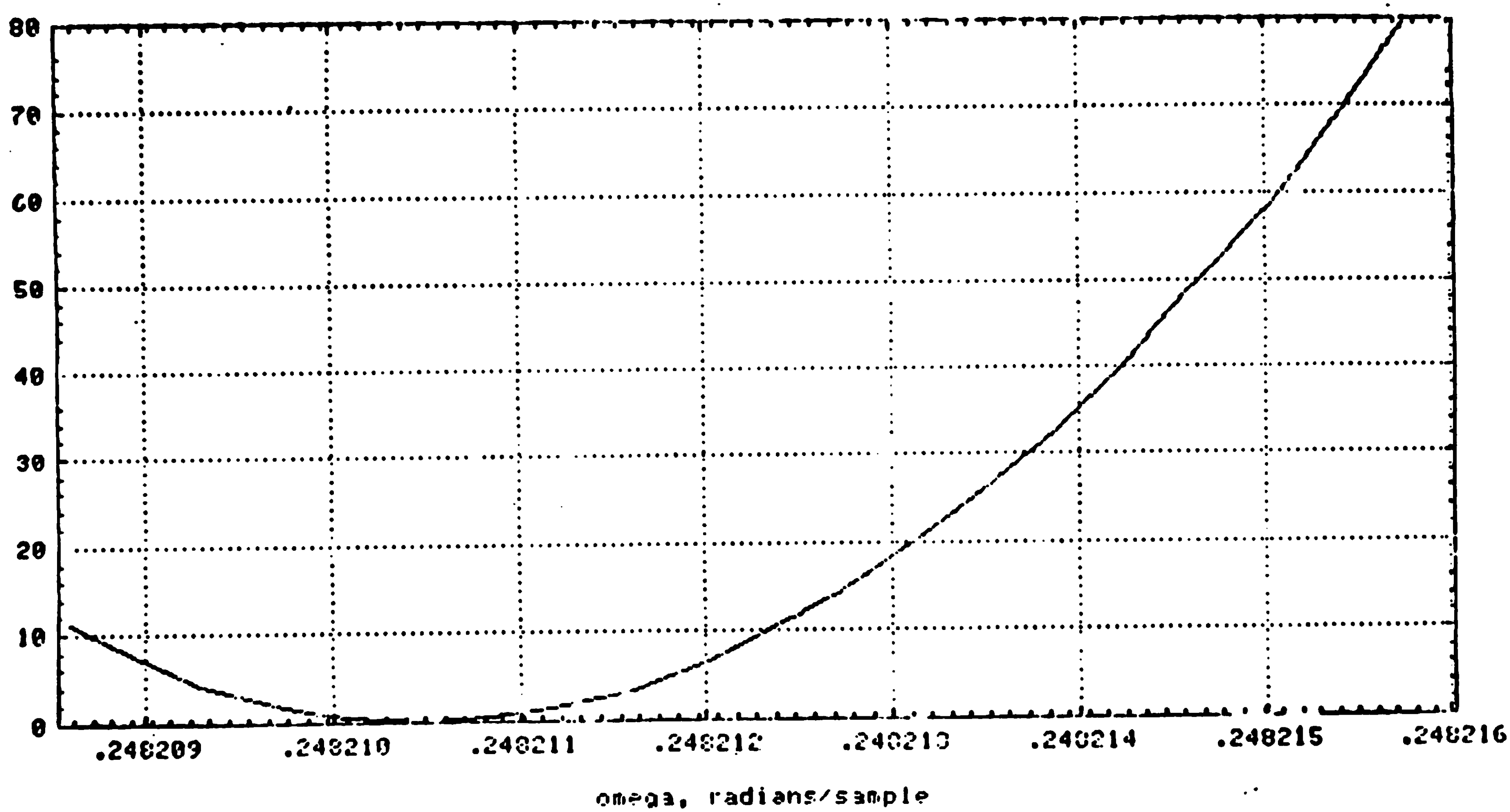

Fig. 2. A plot of $\xi(\omega)$ versus ω for 20 frequency values uniformly spaced within the interval $\left[\omega_1 - \frac{\omega_1}{N},\ \omega_1 + \frac{\omega_1}{N}\right]$ The input data were 64K samples from an ideal 8-bit ADC driven by a sinusoid at a digital frequency of 0.24821044 radians/sample.

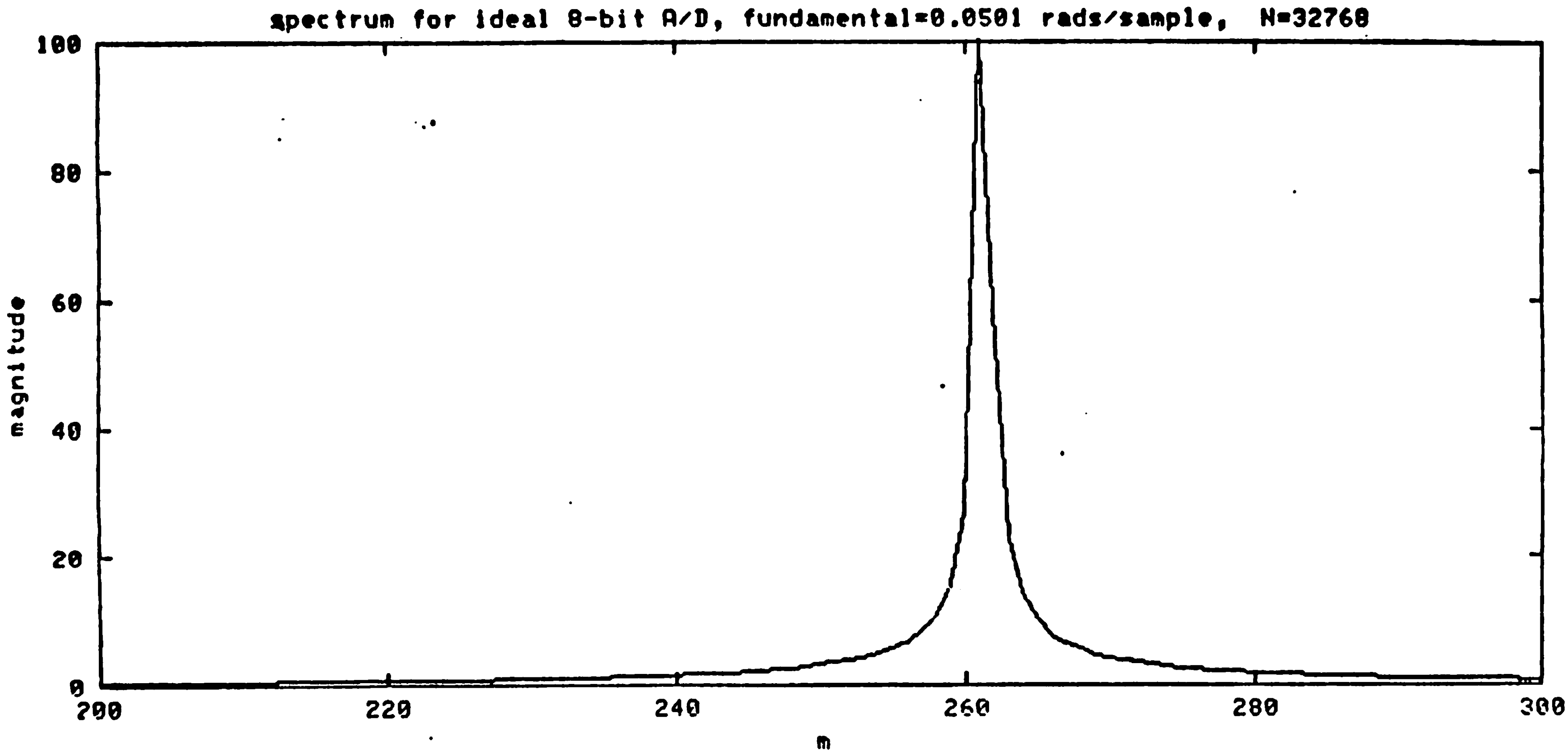

Fig. 3. Magnitude spectrum as calcualted from (3.15) for 32K samples from an ideal 8-bit ADC.

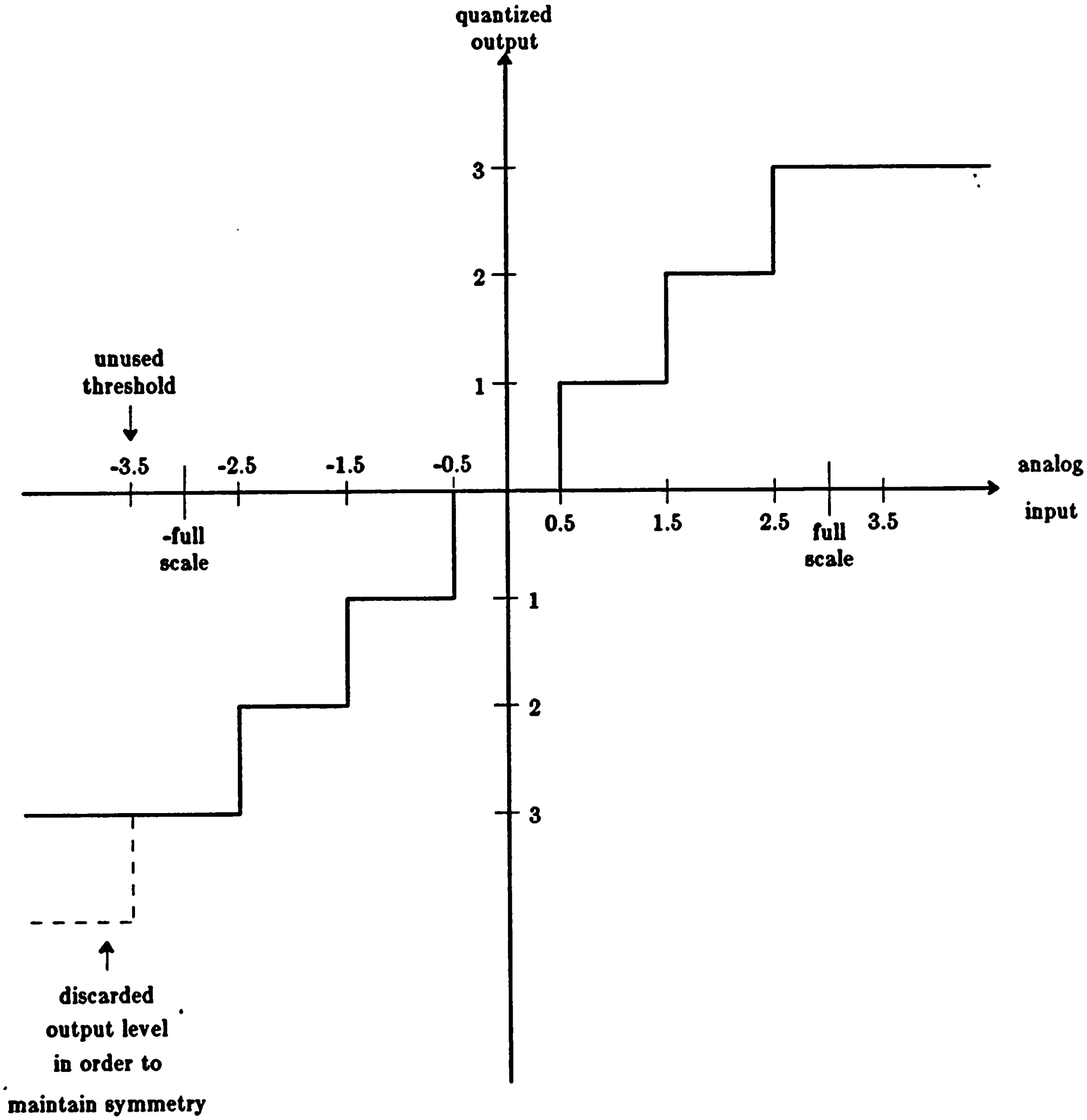

Fig. 4. Transfer characteristics for an ideal 3-bit ADC.

```
12-bit A/D. 500002 samples

dc component of input:  -0.00408400000
total average power of input:  2095105.80914400000

Calculations of initial 3 points for quadratic fit:

w = 0.24820980368
A = 2012.19069951686
B = -324.99680860240
mag = 2077259.11560677850
0.00495113 + 2038.26748412075cos(0.24821n + 0.160131)

mean square error: 17846.69353722146500


w = 0.24821030010
A = 2045.23225325981
B = -73.71286974158
mag = 2094199.49089816200
0.00111575 + 2046.56017672086cos(0.24821n + 0.0360257)

mean square error: 906.31824583798880


w = 0.24821079652
A = 2036.43046404955
B = 179.83339002628
mag = 2089693.38076076670
-0.00283771 + 2044.35541994962cos(0.248211n + -0.0880797)

mean square error: 5412.42838323325850


Repetition 1:
w = 0.24821044401 radians/sample   (25.3139 samples/cycle)
A = 2047.00243220940
B = -0.09980709036
mag = 2095105.72457488990
-2.35002e-05 + 2047.00243464259cos(0.24821n + 4.87577e-05)

mean square error:  0.08456911006942

Repetition 2:
w = 0.24821044425 radians/sample   (25.3139 samples/cycle)
A = 2047.00243444020
B = 0.02298511198
mag = 2095105.72616070190
-2.54064e-05 + 2047.00243456924cos(0.24821n + -1.12287e-05)

mean square error:  0.08298329811078

Repetition 3:
w = 0.24821044420 radians/sample   (25.3139 samples/cycle)
A = 2047.00243476740
B = 0.00033940855
mag = 2095105.72624622670
-2.50546e-05 + 2047.00243476742cos(0.24821n + -1.65806e-07)

mean square error:  0.08289777327445

Repetition 4:
w = 0.24821044420 radians/sample   (25.3139 samples/cycle)
A = 2047.00243476745
B = 0.00033242907
mag = 2095105.72624623020
-2.50547e-05 + 2047.00243476747cos(0.24821n + -1.62398e-07)

mean square error:  0.08289776978199

Repetition 5:
w = 0.24821044420 radians/sample   (25.3139 samples/cycle)
A = 2047.00243477868
B = -0.00139398045
mag = 2095105.72624572910
-2.50279e-05 + 2047.00243477915cos(0.24821n + 6.80986e-07)

mean square error:  0.08289827092085

Minimum mean square error:  0.08289776978199
at repetition 4
```

Fig. 5. Sample program output for an ideal 12-bit converter driven by input 2047cos(0.248210444202n).

Bits	Theoretical Value	Program's Results
4	0.07588	0.07504
8	0.08158	0.08153
10	0.08246	0.08245
12	0.08290	0.08294
14	0.08311	0.08306

Table 1. MMSE values as reported by the sine-fitting program using 100,000 samples from an ideal ADC at the digital frequency 0.24821 radians/sample and theoretical MSE values as calculated from (4.4).

sample rate: 19.532 KHz

analog input	samples/cycle	MMSE	Effective Bits
1.949 KHz	10.02	0.5041	10.70
		0.4555	10.77
1.293 KHz	15.11	0.4344	10.81
		0.4438	10.79
998 Hz	19.57	0.5723	10.61
		0.5837	10.60
651 Hz	30.01	0.3191	11.03
		0.3420	10.98
448 Hz	43.52	0.2461	11.22
		0.2350	11.25
251 Hz	77.82	0.2925	11.09
		0.3124	11.05
91 Hz	196.76	0.1908	11.40
		0.1883	11.41

Table 2. MMSE values from 64K samples of an actual 12-bit bipolar laser-trimmed R-2R ADC.