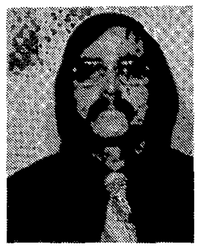
ACKNOWLEDGMENT

The author wishes to express his appreciation to the following people at Bell Laboratories: P. C. Lopiparo and G. Zaccaria for their help with the A/D and D/A converter software drivers; J. R. Cavanaugh, W. C. Kublin, T. A. Pappas, and E. C. Stevens for conducting the subjective tests.

REFERENCES

[1] N. S. Jayant, "Digital Coding of Speech Waveforms: PCM, DPCM, and DM Quantizers," *Proc. IEEE,* Vol. 62, PP. 611-632, May 1974.
[2] I. M. McNair, "Subscriber Loop Multiplexer–A High Pair Gain System for Upgrading and Growth in Rural Areas," *IEEE Trans. Commun. Technol.,* Vol. COM-19, PP. 523-527, Aug. 1971.
[3] R. J. Canniff, "Signal Processing in SLC-40, A 40 Channel Rural Subscriber Carrier," *IEEE ICC 1975 Conference Record,* Vol. 3, June 16-18, 1975, PP. 40-7, 40-11.
[4] C. L. Dammann, L. D. McDaniel, C. L. Maddox, "Multiplexing and Coding," *Bell System Tech. Journal,* Vol. 51, PP. 1675-1700, October 1972.
[5] D. L. Duttweiler, D. G. Messerschmitt, "Nearly Instantaneous Companding for Nonuniformly Quantized PCM," *IEEE Trans. on Commun.,* Vol. COM-24, PP. 864-873, Aug. 1976.
[6] K. L. McAdoo, "Speech Volumes on Bell System Message Circuits, 1960 Survey," *Bell System Tech. Journal,* Vol. 42, PP. 1999-2012, Sept. 1963.

★



William R. Daumer (S'72–M'73) was born in Norwood, Mass., on September 9, 1950. He received the B.Sc. and M.Sc. degrees in Electrical Engineering from Drexel University in 1972 and 1973, respectively.

Mr. Daumer joined Bell Telephone Laboratories, Holmdel, N.J., in 1973, and is currently a staff member in the Digital Network Planning Department. During this time his main concern has been planning for the influx of digital technologies into the evolving switched telecommunications network.

Analysis of Digitally Generated Sinusoids with Application to A/D and D/A Converter Testing

DONALD L. DUTTWEILER, MEMBER, IEEE, AND DAVID G. MESSERSCHMITT, MEMBER, IEEE

***Abstract*–Several aspects of the design of digital frequency synthesizers are considered, and a detailed analysis of the spectral purity of digitally synthesized frequencies is given. These results are then applied to a specific application: the testing of A/D and D/A converters.**

1. INTRODUCTION

FREQUENCY synthesis, which is the process of deriving one frequency from another reference frequency, is best known as a means for obtaining many synchronous sinusoids or clocks. Such signals are required, for example, in communication receivers and digital communications systems. A less well known application of frequency synthesis is to the testing of A/D and D/A converters (codecs) where digitally derived sinusoids are useful. While the work reported here is motivated by the latter application, the results are also applicable to the former.

There are several methods of frequency synthesis,[2,3,4,5] but this paper concentrates on digital frequency synthesis[1,2] in which the samples of a sinusoid are generated digitally and a D/A converter is used to generate the desired sinusoid. In applications of the type first mentioned above the generated sinusoid is the primary objective, while in the second, the objective is to test the operation of the D/A converter itself.

Section 2.0 considers several aspects of the design of digital frequency synthesizers, including an analysis of the spectral properties of the quantization noise. In Section 3.0 we apply these results to the testing of A/D and D/A converters. Appendix A supplies some mathematical details.

Manuscript received October 4, 1977; revised January 26, 1978.
D. L. Duttweiler is with Bell Laboratories, Holmdel, NJ 07733.
D. G. Messerschmitt is with Bell Laboratories, Holmdel, NJ on leave at the University of California, Berkeley, CA 94720.

2.0 DIGITAL FREQUENCY SYNTHESIS

The set of frequencies which can be obtained from a digital frequency synthesizer is considered in Section 2.1, and in Section 2.2 the problem of obtaining the best approximation to a desired frequency with a given synthesizer complexity is solved. Section 2.3 studies several types of symmetries which can be exploited to reduce the memory requirements for stored sinusoids (recursive generation techniques can also be used[1]). Finally, in Section 2.4 the spectrum of the synthesized signal is derived and formulas are given for the SNR.

2.1 Structure

Assume that a sinusoid $A \sin(2\pi f_d t + \theta)$ with amplitude A, frequency f_d and phase θ is desired. We can generate such a sinusoid by storing digital encodings of its samples

$$x(k) = A \sin(2\pi f_d kT + \theta) \tag{2.1}$$

in a read-only-memory (ROM) and presenting them at the rate

$$f_o = 1/T \tag{2.2}$$

for decoding in a D/A converter (see Figure 1).

The output of the D/A converter is a train of impulses

$$y(t) = \sum_{k=-\infty}^{\infty} y(k)\delta(t - kT) \tag{2.3}$$

where

$$y(k) = Q(x(k)) \tag{2.4}$$

and the function $Q(\cdot)$ is the quantization characteristic of the D/A converter*. If $y(t)$ is filtered by an ideal low-pass filter with a cutoff at $f_o/2$ and, as will be assumed from here on, f_d is less than $f_o/2$, the output $w(t)$ of the filter will be the sinusoid $A \sin(2\pi f_d t + \theta)$ plus quantization noise.

To make the memory requirements finite, it is necessary that the samples $x(k)$ repeat with some period N. The condition

$$x(k + N) = x(k) \tag{2.5}$$

for all k is equivalent to

$$f_d NT = L \tag{2.6}$$

for some integer L. Hence the available frequencies are of the form

$$f_d = Lf_o/N. \tag{2.7}$$

Substituting (2.7 into (2.1) gives

$$x(k) = A \sin(2\pi Lk/N + \theta). \tag{2.8}$$

Only the integers

$$L \leqslant [N/2], \tag{2.9}$$

where the square brackets denote the greatest integer less than or equal to the argument, lead to frequencies f_d within the bandwidth $f_o/2$ of the low-pass filter. Furthermore, we need only consider the case for which L and N are relatively prime**, for if the lowest common denominator d of L and N is not 1, then

$$\begin{aligned} x(k + N/d) &= A \sin(2\pi L(k + N/d)/N + \theta) \\ &= A \sin(2\pi Lk/N + 2\pi L/d + \theta) \\ &= A \sin(2\pi Lk/N + \theta) \\ &= x(k) \end{aligned} \tag{2.10}$$

and there is premature repetition of the sequence. Thus, the same frequency Lf_o/N can be generated with less memory by replacing L by L/d and N by N/d.

* In actuality, the quantization is due to the representation of the samples by finite precision arithmetic, and not by the D/A converter. It is convenient to think of the digitally generated samples as having been generated by an ideal A/D converter operating on an ideal analog sinusoid.

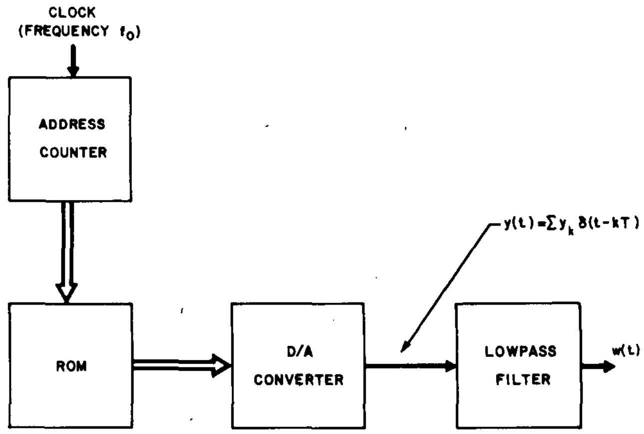


Figure 1. A digital frequency synthesizer.

2.2 Continued Fraction Approximation

If it is important that the digital sine wave have a frequency very close to a desired frequency f^*, continued fraction expansions[6] can be used to find the best approximation $L^* f_o / N^*$ to f^* which can be obtained with N^* constrained to be less than some maximum memory size N_{max}. Rather than develop this theory here, we will refer the reader to reference 6 and give just one example.

For the application motivating our work, $f_o = 8$ kHz and f_d is desired to be as close as possible to 1020 Hz, which is the center frequency of a notch filter. The continued fraction expansion

$$\frac{1020}{8000} = \frac{51}{400} = \cfrac{1}{7 + \cfrac{1}{1 + \cfrac{1}{5 + \cfrac{1}{2 + \cfrac{1}{1 + \cfrac{1}{2}}}}}}$$

** Two integers are relatively prime if they contain no common factors other than unity.

yields when truncated the series of increasingly better approximations of Table 1. The fact that for each of these entries L and N are relatively prime is proved in general in continued fraction theory. It can also be shown that each of these approximations is best in the sense that it gives a better approximation to 1020 Hz than any other rational fraction with a smaller denominator.

2.3 Symmetries to Reduce Memory Requirements

For particular choices of N and θ, there are symmetries in the sample set $\{x(k)\}_{k=0}^{N-1}$ that can be used to reduce memory storage requirements in a ROM implementation.*** The four conceivable forms of symmetry are

$$x(k) = x(k + K_1) \tag{2.11}$$

$$x(k) = -x(k + K_2) \tag{2.12}$$

$$x(k) = x(K_3 - k) \tag{2.13}$$

$$x(k) = -x(K_4 - k). \tag{2.14}$$

Assuming that L and N are relatively prime, (2.11) is satisfied only for $K_1 = N$ and multiples thereof. Of the remaining three symmetries only two are independent, as we show by deriving (2.14) from (2.12) and (2.13). From (2.13)

$$x(k + K_2) = x(K_3 - K_2 - k)$$

and substituting in (2.12)

$$x(k) = -x(K_3 - K_2 - k),$$

which is (2.14) with $K_4 = K_3 - K_2$. Thus there are only two independent forms of symmetry that are potentially useful. Arbitrarily, but without loss of generality, we choose to consider symmetries of the form

$$x(k) = -x(k + K) \tag{2.15}$$

and

$$x(k) = x(K - k). \tag{2.16}$$

An exhaustive list of all symmetries of these forms appears in Table 2. The function mod used in this table is the remaindering function; that is, for two real numbers a and b, a mod b denotes the remainder when a is divided by b.

To exploit symmetry (2.15), it is only necessary to choose N even and read cyclically the samples $\{x(0), \cdots, x(N/2 - 1)\}$ with signs complemented on alternate passes. To exploit the fourfold symmetry of (2.15) and (2.16) together, it is necessary to choose N even and θ equal to either 0 mod $2\pi/N$ or π/N mod $2\pi/N$.

*** Similar symmetries have been used to reduce computation in DFT and FFT analysis where they are more readily apparent because the phase is always zero.

TABLE 1
BEST CHOICES FOR N AND L

N	L	f_d (Hz)
8	1	1000.00
47	6	1021.27
102	13	1019.61
149	19	1020.13
400	51	1020.00 (exact)

TABLE 2
EXHAUSTIVE LIST OF SYMMETRIES

Type of Symmetry	N	θ	K Uniquely Satisfies
(2.15)	even	Arbitrary	$K = N/2$
(2.16)	even	0 mod $2\pi/N$	$K = N/2$
(2.16)	even	π/N mod $2\pi/N$	$\left\{LK = \frac{N-2}{2}\right\}$ mod N
(2.16)	odd	$\pi/2N$ mod $2\pi/N$	$\left\{LK = \frac{N-1}{2}\right\}$ mod N
(2.16)	odd	$3\pi/2N$ mod $2\pi/N$	$\left\{LK = \frac{N-3}{2}\right\}$ mod N

For an example of the application of fourfold symmetry, assume N is even, $N/2 \triangleq K$ is even, and θ equals 0 mod $2\pi/N$. Then (2.15), (2.16), and Table 2 indicate that

$$\begin{array}{lllllll} x(0) & & & = & -x(K) & & \\ x(1) & = & x(K-1) & = & -x(K+1) & = & -x(N-1) \\ \vdots & & \vdots & & \vdots & & \vdots \\ x(K/2-1) & = & x(K/2+1) & = & -x(3K/2-1) & = & -x(3K/2+1) \\ x(K/2) & & & = & -x(3K/2). & & \end{array} \tag{2.17}$$

Thus, it is sufficient to store the $N/4 + 1$ samples $\{x(0), \cdots, x(K/2)\}$ in memory, address them with a counter counting cyclically up from 0 to $K/2$ and then down from $K/2$ to 0, and complement signs on alternate cycles.

2.4 Signal-to-Noise Ratio (SNR)

The D/A output of Figure 1 is periodic with period NT and thus has the Fourier series representation

$$y(t) = (1/T) \sum_{m=-\infty}^{\infty} Y(m) e^{j2\pi m t f_0/N} \tag{2.18}$$

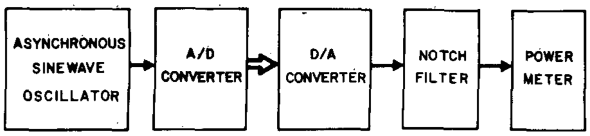


Figure 2. Traditional method of testing A/D and D/A converters with an asynchronous sinewave.

where

$$Y(m) = (1/N) \sum_{k=0}^{N-1} y(k) e^{-j2\pi mk/N}. \tag{2.19}$$

We will assume the LPF to have the transfer function†

$$H(f) = \begin{cases} T, & |f| < f_o/2 \\ T/\sqrt{2}, & |f| = f_o/2 \\ 0, & |f| > f_o/2. \end{cases} \tag{2.20}$$

The issue of just what $H(f)$ is to be at $f_o/2$ is forced by the line spectrum nature of the input. The above choice greatly simplifies later equations. It is artificial from a practical point of view in the same way in which any other ideal low-pass filter is artificial.

The filter output is then

$$w(t) = \begin{cases} \displaystyle\sum_{m=-(N-1)/2}^{(N-1)/2} Y(m) e^{j2\pi m t f_o/N}, & N \text{ odd} \\ \displaystyle\sum_{m=-N/2+1}^{N/2-1} Y(m) e^{j2\pi m t f_o/N} \\ \quad + (1/\sqrt{2})[Y(N/2)e^{j\pi t f_o} \\ \quad + Y(-N/2)e^{-j\pi t f_o}], & N \text{ even.} \end{cases} \tag{2.21}$$

This output contains the desired component $Y(L)$ at frequency Lf_o/N, while the other components in the spectrum are undesired spurious responses. The equation demonstrates the harmonic nature of the synthesizer output, there being all harmonics of f_o/N.

The ratio of the power at the desired frequency to the total power at all other frequencies except d.c., which it is natural to call the signal-to-noise ratio (SNR), is given by

$$\text{SNR} = \begin{cases} |Y(L)|^2 \Big/ \left(\displaystyle\sum_{\substack{m=1 \\ m \neq L}}^{(N-1)/2} |Y(m)|^2 \right), & N \text{ odd} \\ |Y(L)|^2 \Big/ \left(\displaystyle\sum_{\substack{m=1 \\ m \neq L}}^{N/2-1} |Y(m)|^2 \right. \\ \quad \left. + (1/2)\,|Y(N/2)|^2 \right), & N \text{ even.} \end{cases} \tag{2.22}$$

† Note that we could improve the SNR derived later by letting $H(f)$ be a narrow bandpass filter centered at f_d, if f_d is a fixed frequency. Choice (2.20) is motivated by our application in which the D/A converter is tested by using it in the mode of Figure 1, but it also is appropriate for other purposes.

By straightforward manipulation, it can be shown that two other equivalent formulas are

$$\text{SNR} = 2\,|Y(L)|^2 \Big/ \left[\frac{1}{N} \sum_{k=0}^{N-1} \left(y(k) - Y(0) - 2\,\text{Re}\,(Y(L)e^{j2\pi Lk/N})\right)^2 \right] \tag{2.23}$$

and

$$\text{SNR} = 2\,|Y(L)|^2 \Big/ \left[\frac{1}{N} \sum_{k=0}^{N-1} y^2(k) - Y^2(0) - 2\,|Y(L)|^2 \right]. \tag{2.24}$$

The advantage in using either of the latter two relations is that the only Fourier coefficients that need to be calculated are $Y(L)$ and $Y(0)$. Furthermore, (2.23) is intuitively satisfying since the signal component in $w(t)$ is

$$Y(L)e^{j2\pi Lk/N} + Y^*(L)e^{-j2\pi Lk/N} = 2\,\text{Re}\,(Y(L)e^{j2\pi Lk/N}), \tag{2.25}$$

while (2.24) is also intuitive since the quantization noise is the total power less the power in the d.c. and f_d components.

It is interesting to note that the SNR as defined by any of the equivalent relations (2.22), (2.23), and (2.24) is independent of L as long as L and N are relatively prime. Changing the frequency of the desired sinewave (through L) does not change the total power in the spurious responses; rather, it results in a permutation of the same spectral lines. Proofs of these facts are given in Appendix A, which also contains other relationships of interest in their own right.

3.0 A/D AND D/A CONVERTER TESTING

The traditional method of testing an A/D and D/A converter in cascade is shown in Figure 2. The input to the A/D converter is a sinusoid asynchronous with the sampling clock. By passing the output of the D/A through a notch filter rejecting the frequency of the input sinusoid, the fundamental in the D/A output can be removed. The power in the remaining signal is the quantization noise power.

An undesirable aspect of the above procedure is that faults cannot be isolated to either the A/D or D/A. This problem is usually more troublesome when it is desired to test the D/A alone, since its performance is usually closer to ideal than that of the A/D. It is therefore natural to ask whether digital signal processing techniques can be substituted for portions of Figure 2, and whether the D/A and A/D can be tested individually by this means. We consider these questions in the following three sections.

3.1 D/A Converter Testing

To test a D/A alone, static output level measurements could be made. Alternately and generally preferably, however, the

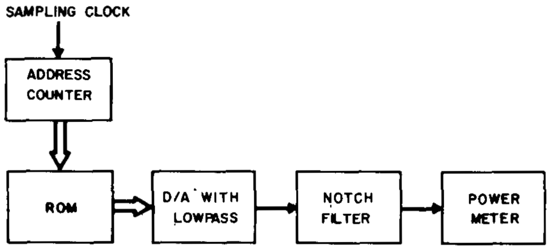


Figure 3. Testing a D/A converter with a digital sinewave.

test structure of Figure 3 can be used.†† The structure here less the notch filter and power meter is exactly that of the digital frequency synthesizer considered in Section 2.0.

The ROM in Figure 3 is equivalent to an ideal A/D driven by a sinusoid synchronous with the sampling clock (see Figure 4). Thus, if the sinusoid were not constrained to be synchronous, we would effectively have available an ideal A/D for testing D/A converters.

It is desirable that asynchronous sinusoids be used for coder testing because (i) specifications for coders have been written assuming asynchronous sinusoids and (ii) designers have developed an intuition for relating SNR measurements made with asynchronous sinusoids to coder imperfections. Asynchronous sinusoids exercise all decoder levels less than a peak level, and, moreover, the relative frequencies of the levels are fixed (by the probability density function of a sinusoid). With a low N digitally generated sinusoid, not all the output levels less than the peak level are exercised, and, of course, the relative exercise proportions are not proper. The SNR measured for a low N digital sinusoid is thus not necessarily equal or even close to the SNR measured for an equal amplitude asynchronous sinusoid.

We have attempted to answer the questions "how large must N be for a synchronous sinusoid to look like an asynchronous sinusoid for coder testing purposes?" and "can the symmetries of Section 2.3 be advantageously exploited?" The procedure used has been to calculate $\mathrm{SNR}_S(A, \theta, N)$ as given by equation (2.23) (either (2.22) or (2.24) could equivalently have been used) for various sinusoid amplitudes A, sinusoid phases θ, and memory periods N and compare it with $\mathrm{SNR}_A(A)$, the signal-to-noise ratio for an asynchronous sinusoid of amplitude A. In all these calculations an ideal D/A and 8-bit μ255 encoding have been assumed. Presumably, the results for some particular nonideal D/A or some other encoding would not be substantially different (with one exception to be noted later).

The calculations indicate that with N greater than about 100 the difference between SNR (A, θ, N) and $\mathrm{SNR}_A(A)$ is less than 0.5 dB as long as θ and N are such that neither of the symmetries (2.15) or (2.16) is present. If θ and N are such that twofold symmetry is present, N must be greater than about 200 for equivalent accuracy. With fourfold symmetry present, N must be greater than about 400.

Thus the general conclusion to be drawn at this point appears to be that there is nothing to be gained by choosing N and θ so that there are sample set symmetries to be exploited. Performance is proportional to the minimum memory actually required.

†† This technique has been used in PCM channel banks for some time[8].

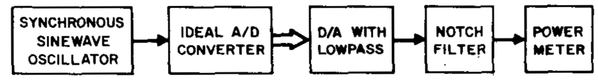


Figure 4. Equivalent block diagram for the D/A converter test.

We do not feel, however, that this conclusion is completely correct for a nonideal D/A, for which there is probably some advantage to be gained by choosing N even and exploiting symmetry (2.15). The reason is that although with an ideal D/A the quantization error associated with a sample $x(k)$ is identical (in magnitude) to the quantization error associated with the sample $-x(k)$, with a nonideal D/A it is not. By choosing N even and exploiting symmetry (2.15), the nonideal D/A is more completely tested. This same argument does not, however, support exploiting symmetry (2.16).

3.2 A/D Converter Testing

When an A/D and D/A are tested as in Figure 2, any fault can usually be traced to either the A/D or D/A by the simple procedure of Section 3.1. Nevertheless, for completeness we should mention that the A/D converter alone can also be tested by the somewhat more complex method of Figure 5.††† The synchronous sinusoid can be obtained by phase-locking an oscillator to the same clock which controls the sampling in the A/D. The SNR can then be calculated digitally from a sequence of N samples at the A/D output using (2.22), or preferably (2.23) or (2.24). It is important to note that these SNR formulas do not require prior knowledge of the sinusoid amplitude, which is important from a practical viewpoint since this amplitude is difficult to control precisely in the analog portion of the test. The conclusions in Section 3.1 about the required N also apply directly to this case.

3.3 A/D and D/A Converter Testing

As an alternative to the method of Figure 2, the features of Figures 3 and 5 can be combined (Figure 6) to achieve testing of both an A/D and D/A converter together. As in Figure 2, it is not possible to distinguish noise introduced by the A/D or D/A alone. The advantages of this approach are the ease of automation and reproducibility of the test, as well as in some application a more natural interface on the digital computer with A/D and D/A interfaces.

We should note that in Figure 6 there are three sources of quantization noise: a) the noise produced by the initial finite precision representation of the sinusoid, b) noise introduced by nonideal behavior of the D/A, and c) noise produced by the A/D. In Figure 2 the only sources of noise are b) and c), and hence Figure 6 will yield a calculated noise larger than that measured in Figure 2 by the amount of noise a).§ However, this should present no practical difficulty, since the noise of a) is not variable and can be precisely calculated beforehand and discounted in the interpretation of the results.

4.0 CONCLUSION

It is hoped that this paper will provide an enhanced understanding of digitally generated sinusoids and particularly their

††† This structure has been previously described in reference 9.

§ In fact, if the A/D and D/A were ideal, the SNR calculated in Figure 6 would be that of two tandem A/D conversions, or 3 dB lower than that of a single conversion.

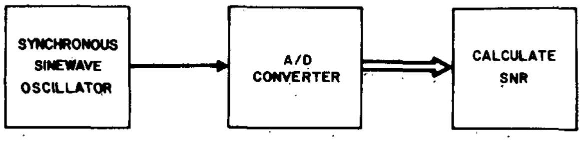


Figure 5. Testing an A/D converter.

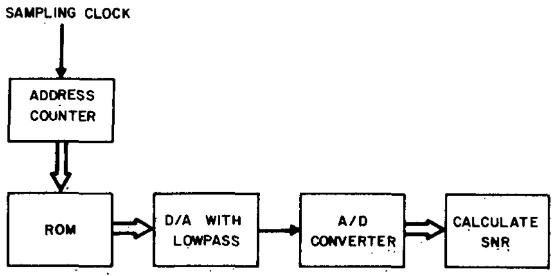


Figure 6. Digital testing of A/D and D/A converters.

application to A/D and D/A converter testing. Related ideas can be applied to phase-locked loop frequency synthesis to obtain an interesting new structure.[10]

APPENDIX A

In this appendix we wish to show that the signal-to-noise ratio of a digital sinusoid is independent of L as long as L and N are relatively prime. Since it will be important to have a notation explicitly indicating dependence on L we will adopt the convention of using subscripts to show this dependence. Thus, rewriting (2.19), we have

$$Y_L(m) = (1/N) \sum_{k=0}^{N-1} y_L(k) e^{-j2\pi mk/N} \tag{A.1}$$

where

$$y_L(k) = y_1(kL \bmod N). \tag{A.2}$$

Although the following results apply in greater generality, in Section 2.0 we have the particular case

$$y_1(k) = Q(x_1(k)) \tag{A.3}$$

$$x_1(k) = A \sin(2\pi k/N + \theta). \tag{A.4}$$

Define the set of numbers

$$A_L = \{y_L(k)\}_{k=0}^{N-1}. \tag{A.5}$$

Then an interesting fact is that $A_L = A_1$ for all L such that L and N are relatively prime. Of course A_L is contained in A_1 from (A.2). The converse follows from the existence of a unique integer $0 \leqslant J < N$ satisfying the relation§§

$$LJ \bmod N = 1. \tag{A.6}$$

§§ This is a fundamental result of number theory[6] which requires that L and N be relatively prime. In a sense J is the multiplicative inverse of L.

We can use (A.6) to find the inverse mapping to (A.2) by defining the new index

$$q = kL \bmod N \tag{A.7}$$

and noting that

$$\begin{aligned} qJ \bmod N &= J(kL \bmod N) \bmod N \\ &= kJL \bmod N. \end{aligned} \tag{A.8}$$

Substituting from (A.6), (A.8) becomes

$$k = qJ \bmod N. \tag{A.9}$$

Thus (A.2) becomes

$$y_1(q) = y_L(qJ \bmod N) \tag{A.10}$$

and the fact that A_1 is contained in A_L is demonstrated.

The discrete Fourier transform (DFT) of the sequence $y_L(k)$ is given by (2.19) and (A.2), that is,

$$Y_L(m) = \frac{1}{N} \sum_{k=0}^{N-1} y_1(kL \bmod N) e^{-j2\pi mk/N}. \tag{A.11}$$

Reindexing (A.11) using (A.7) and (A.9), we have

$$\begin{aligned} Y_L(m) &= \frac{1}{N} \sum_{q=0}^{N-1} y_1(q) e^{-j2\pi m(qJ \bmod N)/N} \\ &= \frac{1}{N} \sum_{q=0}^{N-1} y_1(q) e^{-j2\pi q(mJ \bmod N)/N} \\ &= Y_1(mJ \bmod N). \end{aligned} \tag{A.12}$$

Equation (A.12) establishes that the DFT coefficients undergo a rearrangement of the same type as the $\{y_L(k)\}$ as L is varied. The fact that the set of DFT coefficients $\{Y_L(m)\}_{M=0}^{N-1}$ is the same as the set $\{Y_1(m)\}_{M=0}^{N-1}$ follows from the fact that N and J are relatively prime.

We are now in a position to show that the SNR of (2.22) is independent of L. First note that from (A.12)

$$Y_L(L) = Y_1(LJ \bmod N) = Y_1(1) \tag{A.13}$$

and $Y_L(0) = Y_1(0)$. Hence, when N is odd

$$\sum_{\substack{m=1 \\ m \neq L}}^{(N-1)/2} |Y_L(m)|^2 = \sum_{m=2}^{(N-1)/2} |Y_1(m)|^2 \tag{A.14}$$

and the SNR is independent of L. For N even we need the additional fact that

$$\begin{aligned} Y_L(N/2) &= Y_1((N/2)J \bmod N) \\ &= Y_1(N/2), \end{aligned} \tag{A.15}$$

which follows from the fact that J must be odd if N is even.

Finally we should note that in reference 7 relation (A.2) was called the "uniform permutation" of samples $\{y_1(k)\}_{k=0}^{n-1}$. The fact that the DFT coefficient $\{Y_1(m)\}_{m=0}^{n-1}$ also undergo a uniform permutation was discovered independently in [7].

REFERENCES

1. J. Tierney, C. M. Rader, and B. Gold, "A Digital Frequency Synthesizer," *IEEE Trans. Audio and Electroacoustics,* Volume AU-19, p. 48 (1971).
2. J. Gorski-Popiel, Editor, *Frequency Synthesis–Techniques and Applications,* New York: IEEE Press, 1975.
3. J. Noordanus, "Frequency Synthesizers–A Survey of Techniques," *IEEE Trans. Comm. Tech.,* Volume COM-17, p. 257, April, 1969.
4. V. F. Kroupa, *Frequency Synthesis,* New York: Wiley, 1973.
5. V. Manassewitsch, *Frequency Synthesizers Theory and Design,* New York: Wiley, 1976.
6. B. M. Stewart, *Theory of Numbers,* New York: Macmillan, 1964.
7. S. C. Kak and N. S. Jayant, "On Speech Encryption Using Waveform Scrambling," *BSTJ,* Volume 56, May-June, 1977, p. 781.
8. H. H. Henning and J. W. Pan, "D2 Channel Bank: System Aspects," *BSTJ,* Volume 51, October, 1972, p. 1641.
9. W. Pratt, "Test A/D Converters Digitally," *Electronic Design,* December 6, 1975, p. 86.
10. D. G. Messerschmitt, "A New PLL Frequency Synthesis Structure," submitted to *IEEE Transactions on Communications.*

★

Donald L. Duttweiler (S'68–M'70) for a photograph and biography see page 653 of this issue.

★

David G. Messerschmitt (S'65–M'68) is an Assistant Professor of Electrical Engineering and Computer Science at the University of California, Berkeley, on leave from Bell Laboratories, Holmdel, New Jersey. Since 1968 he has done research, systems engineering, and exploratory development of digital communication systems, including lines and terminals. From 1974-77 he was Supervisor of a group studying bit rate reduction techniques using digital signal processing. BS, University of Colorado, MS, PhD, University of Michigan. Member, Tau Beta Pi, Eta Kappa Nu, Sigma Xi.

Simple Inductorless Automatic Line Equalizer for PCM Transmission Using New Variable Transfer Function

YOSHITAKA TAKASAKI, MEMBER, IEEE

Abstract—A new variable transfer function is proposed and its application to digital transmission is investigated.

The realization of the new variable transfer function is developed for use in a digital transmission automatic line build out (ALBO) network. A simple inductorless ALBO network having ±15 dB variable range is demonstrated for use in 1.544 Mbit/s PCM transmission. Intersymbol interference and jitter increments due to equalization errors are around ten percent and ±2 degrees, respectively.

I. INTRODUCTION

DIGITAL transmission lines can employ automatic line build out (ALBO) networks owing to their relatively small susceptibility to equalization errors [1]. However, equalization errors have to be made as small as possible from the standpoint of reducing timing jitter [2].

For realizing sufficiently small equalization error over the ALBO networks' wide variable range, cascading of two variable equalizers has been used in conventional PCM repeaters. For example, cascading of two Tarbox type variable equalizers [1] has been reported [3] for realizing a wide variable range. The Bode type variable equalizer [4] may potentially be suitable for realizing ALBO with a single stage. However, realization of one-port dual shaping networks has been the bottleneck for attaining a wide variable range. Equalizer design has also been quite cumbersome in conventional ALBO's. A solution to those problems has been suggested in a previous paper [5].

A simple wide range inductorless variable equalizer for digital transmission is investigated in this paper, as an extension of results reported in [5]. It will be shown that the design procedure is quite straightforward and flexible. Application to a practical system will also be demonstrated.

Manuscript received September 6, 1977; revised January 6, 1978. This paper was presented at the International Conference on Communications, Chicago, IL, June 1977.

The author is with the Central Research Laboratory, Hitachi, Ltd., Kokubunki, Tokyo 185, Japan.

II. NEW VARIABLE EQUALIZERS

The ideal variable function is given by

$$v_0(f, u) = y_0(f)^u, \tag{1}$$

0090-6778/78/0500-0675$00.75