A Sine Generation Algorithm for VLSI Applications

John W. Gordon
Schlumberger Palo Alto Research
3340 Hillview Avenue
Palo Alto, CA 94304
(415) 496–4608

and

Julius O. Smith
CCRMA/Dept. of Music
Stanford University
Stanford, CA 94305
(415) 497–4971

Abstract

Three different approaches to the digital synthesis of sinusoids are examined as to their application in VLSI. Recursive techniques are favored due to their low memory requirements and relatively small computational cost. A second-order (resonator) method is contrasted with a coupled first-order form in regard to bit precision, frequency resolution, amplitude stability, and distortion. The coupled form is found to be more reliable, especially for short word lengths, and is also found to have more promise in being extensible to realize the frequency modulation synthesis technique.

1. INTRODUCTION

Methods for generating sine waves can be grouped into three categories: (a) computation of $\sin(\omega)$ by means of series expansion, where ω can be any arbitrary angle, (b) table lookup of the sine function, interpreting the angle ω as a table address and quantizing it according to table length, and (c) recursive techniques, which cannot compute the sine of an arbitrary angle, but rather compute sequences of samples based on the values of state variables. Methods belonging to category (a) are generally employed in subroutine packages for high-level computer languages, but are usually considered too time- and resource-intensive for musical synthesis purposes. Almost all sine generation algorithms employed for music, whether implemented in hardware or software, make use of the table lookup method. Resulting sinewaves vary in frequency resolution, signal-to-quantization-noise-ratio, and distortion depending on the length and width of the table, and whether or not interpolation is used to smooth the output.

If sine generation is implemented in VLSI, the table lookup method is probably inappropriate, since memory is usually not available or very limited. Method (a) is too expensive unless a very gross approximation to the sine function (such as a cubic) is sufficient; for instance, such an approximation might be satisfactory for FM synthesis. In general, however, recursive techniques must be used. One such technique makes use of a digital resonator (2nd-order filter) with the radius of the poles set to 1.0. With careful control over the filter zeros, the initial conditions, and computation precision, this technique yields stable and accurate sinewaves, and has been used with success by Wawrzynek and Mead (1985) at Caltech.

Another recursive technique uses a coupled first-order filter [Rabiner and Gold (1975)]; rotation of a vector by ω is accomplished by multiplying the vector by an appropriate matrix. If the traditional rotation matrix is used, the amplitude of the resulting sinewave tends to grow or decay exponentially, due to quantization errors in representing the $\cos(\omega)$ and $\sin(\omega)$ coefficients. However, a different matrix can be used that guarantees stability. This alternate method will be referred to as the ***modified coupled first-order form***, and is treated in more detail in the following section. After this discussion, the modified coupled form and the digital resonator are compared in regard to their relative merits as sinusoidal generators in VLSI.

2. OVERVIEW OF COUPLED FORM

The coupled first-order form can be expressed as a pair of equations:

$$x(n+1) = \cos(\omega) \cdot x(n) - \sin(\omega) \cdot y(n)$$
$$y(n+1) = \sin(\omega) \cdot x(n) + \cos(\omega) \cdot y(n),$$

or in matrix form as

$$\begin{aligned} \begin{bmatrix} x(n+1) \\ y(n+1) \end{bmatrix} &= \begin{bmatrix} \cos(\omega) & -\sin(\omega) \\ \sin(\omega) & \cos(\omega) \end{bmatrix} \cdot \begin{bmatrix} x(n) \\ y(n) \end{bmatrix} \\ &= \begin{bmatrix} \cos(\omega) & -\sin(\omega) \\ \sin(\omega) & \cos(\omega) \end{bmatrix}^n \cdot \begin{bmatrix} x(0) \\ y(0) \end{bmatrix} \end{aligned} \qquad (1)$$

It can be shown that this is equivalent to

$$\begin{bmatrix} x(n+1) \\ y(n+1) \end{bmatrix} = \begin{bmatrix} \cos(n\omega) & -\sin(n\omega) \\ \sin(n\omega) & \cos(n\omega) \end{bmatrix} \cdot \begin{bmatrix} x(0) \\ y(0) \end{bmatrix}$$

and thus with the initial condition of $\{x(0),\ y(0)\} = \{1,\ 0\}$, we see that $x(n+1) = \cos(n\omega)$ and $y(n+1) = \sin(n\omega)$. There is numerical instability in this method, arising from inaccurate representation in finite word-length machines of $\cos(\omega)$ and $\sin(\omega)$. More specifically, if $\sim\cos(\omega)$ and $\sim\sin(\omega)$ are used to notate these slight inaccurate representations, the determinant of the matrix, or $\sim\cos^2(\omega) + \sim\sin^2(\omega)$, is not necessarily equal to 1; thus, in general, equation (1) results in waveforms that either decay rapidly into silence or grow exponentially in amplitude, causing nonlinear overflow oscillations.

2.1 The Modified Coupled Form

It is possible to use a slightly different set of coefficients to produce absolutely stable sinusoidal waveforms:

$$\begin{aligned} x(n+1) &= x(n) - \varepsilon \cdot y(n) \\ y(n+1) &= \varepsilon \cdot x(n+1) + y(n), \end{aligned}$$

which in matrix form becomes

$$\begin{bmatrix} x(n+1) \\ y(n+1) \end{bmatrix} = \begin{bmatrix} 1 & -\varepsilon \\ \varepsilon & 1-\varepsilon^2 \end{bmatrix}^n \cdot \begin{bmatrix} x(0) \\ y(0) \end{bmatrix} \qquad (2)$$

A more compact way of expressing (2) is as:

$$\mathbf{x_n} = \mathbf{G^n} \cdot \mathbf{x}_0,$$

where $\mathbf{x_k}$ represents an output vector at sample time k and $\mathbf{G}$ is the modified rotation matrix. It is evident from (2) that the determinant of the matrix is equal to 1 regardless of the fixed-precision value given to ε; hence, $x(n)$ and $y(n)$ should be numerically stable. However, we need to resort to linear systems analysis to verify this, and also to determine the forms of $x(n)$ and $y(n)$ and how they relate to each other.

We can start by finding the eigenvalues (natural, or resonating, frequencies) of $\mathbf{G}$. This is done by solving

$$\det(\lambda\mathbf{I} - \mathbf{G}) = \begin{vmatrix} \lambda - 1 & \varepsilon \\ -\varepsilon & \lambda - 1 + \varepsilon^2 \end{vmatrix} = 0$$

which yields

$$\lambda = 1 - \frac{\varepsilon^2}{2} \pm j\varepsilon\sqrt{1-\varepsilon^2/4},$$

where j is used to represent $\sqrt{-1}$. This formula for λ is valid for $|\varepsilon| < 2$. Since the squared magnitude of λ equals 1, we can represent λ as

$$\lambda = e^{\pm j\omega} = \cos(\omega) \pm j\sin(\omega),$$

where $\cos(\omega) = 1 - \varepsilon^2/2$ and $\sin(\omega) = \varepsilon\sqrt{1-\varepsilon^2/4}$. For convenience, we shall choose $\lambda_1 = e^{j\omega}$ and $\lambda_2 = e^{-j\omega}$. It can also be shown that ε and the natural frequency ω are related by

$$\sin\frac{\omega}{2} = \frac{\varepsilon}{2} \qquad \text{and} \qquad \cos\frac{\omega}{2} = \sqrt{1-\varepsilon^2/4}$$

and that an ε ranging between ± 2 corresponds to an ω ranging between $\pm\pi$.

In order to represent $\mathbf{G^n}$ in terms of ω, we seek to express $\mathbf{G}$ in terms of $\mathbf{\Lambda}$, where $\mathbf{\Lambda} = \left[\begin{smallmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{smallmatrix}\right]$. Since the eigenvalues are distinct (i.e., $\lambda_1 \neq \lambda_2$), there is a transformation matrix $\mathbf{T}$ such that $\mathbf{G} = \mathbf{T\Lambda T^{-1}}$; furthermore, $\mathbf{G^n} = \mathbf{T\Lambda^n T^{-1}}$ (since $\mathbf{T^{-1}T} = \mathbf{I}$), and $\mathbf{\Lambda^n} = \left[\begin{smallmatrix} \lambda_1^n & 0 \\ 0 & \lambda_2^n \end{smallmatrix}\right] = \left[\begin{smallmatrix} e^{jn\omega} & 0 \\ 0 & e^{jn\omega} \end{smallmatrix}\right]$. A $\mathbf{T}$ that satisfies these equations is

$$\mathbf{T} = \begin{bmatrix} 1 & 1 \\ e^{-j\varphi} & e^{j\varphi} \end{bmatrix},$$

where $e^{j\varphi} = \varepsilon/2 + j\sqrt{1-\varepsilon^2/4}$. (Note that the right-hand side of this equation does indeed have a magnitude of 1.) Further investigation shows that $\varphi = (\pi - \omega)/2$; thus, $\cos(\varphi) = \sin(\omega/2) = \varepsilon/2$ and $\sin(\varphi) = \cos(\omega/2) = \sqrt{1-\varepsilon^2/4}$. Using this $\mathbf{T}$ matrix, we have

$$\begin{aligned} \mathbf{G^n} &= \mathbf{T\Lambda^n T^{-1}} \\ &= \frac{1}{2j\sin(\varphi)} \begin{bmatrix} 1 & 1 \\ e^{-j\varphi} & e^{j\varphi} \end{bmatrix} \begin{bmatrix} e^{jn\omega} & 0 \\ 0 & e^{-jn\omega} \end{bmatrix} \begin{bmatrix} e^{j\varphi} & -1 \\ -e^{-j\varphi} & 1 \end{bmatrix} \\ &= \frac{1}{\sin(\varphi)} \begin{bmatrix} \sin(n\omega + \varphi) & -\sin(n\omega) \\ \sin(n\omega) & -\sin(n\omega - \varphi) \end{bmatrix} \end{aligned} \qquad (3)$$

2.2 Initial Conditions

From (3), it is more apparent how to set initial conditions (the vector $\mathbf{x}_0$) to obtain our desired sinusoids. In particular, setting $\{x(0),\ y(0)\} = \{1,\ \cos(\varphi)\} = \{1,\ \varepsilon/2\}$ yields $\{x(n),\ y(n)\} = \{\cos(n\omega),\ \cos(n\omega - \varphi)\}$ and $\{x(0),\ y(0)\} = \{0,\ -\sin(\varphi)\}$ yields $\{x(n),\ y(n)\} = \{\sin(n\omega),\ \sin(n\omega - \varphi)\}$.

3. COMPARISON WITH DIGITAL RESONATOR

A digital resonator is a two-pole recursive filter that can be expressed as

$$y(n) = 2R\cos(\omega) \cdot y(n-1) - R^2 y(n-2) + x(n) \qquad (4)$$

If $R = 1$, the resonator is undamped, and the filter response $y(n)$ to an impulsive input $x(n)$ is a sinusoid with constant amplitude. The phase of the sinusoid can be controlled by using an input sequence different from a single impulse, or by introducing feed-forward components into the filter. That is, a pure sine wave is had if $x(n)$ in equation (4) is replaced by $R\sin(\omega) \cdot x(n-1)$, and a pure cosine wave is obtained by replacing $x(n)$ with $x(n) - R\cos(\omega) \cdot x(n-1)$. Without the feedforward terms, the impulse $x(0)$ must be prescaled by $\sin(\omega)$ to guarantee an output sinusoid bounded by ± 1.

It is important to realize that both the digital resonator and the modified coupled form have coefficients that can range between ± 2. For the resonator, the coefficient in question is $2\cos(\omega)$ (R is 1); for the coupled form, it is ε. This means that to implement either of these methods requires a number system that can accommodate values in this range. (The critical issue is how results are obtained from multiplications.) Both the Caltech system [Wawrzynek and Mead (1985)] and the MSSP [Lyon (1984)] use numbers that are interpreted as integer-fractions with the binary point positioned such that the operating range is between ± 8.

Both sine generation methods were simulated at the Center for Computer Research in Music and Acoustics (CCRMA) at Stanford. Word length (number of bits of precision), frequency, and duration were treated as parameters, so that amplitude stability, frequency resolution, and distortion could be examined under different operating conditions. The sampling rate was 44.1 KHz in all examples, fixed-point arithmetic was used throughout, and simple truncation was used to trim values to the set word length after multiplications. (Rounding and truncation towards 0 were also tried, but results were essentially identical to those obtained using simple truncation.)

3.1 Computational Cost

With R set to 1, the resonator involves one multiply and one add per output sample. In addition, a memory move or copy is required so that $y(n-2)$ gets the old $y(n-1)$ before $y(n-1)$ is overwritten with the new $y(n)$. No such move is required for the modified coupled form, since $x(n+1)$ is updated first, and then used with the old $y(n)$ to calculate $y(n+1)$. However, two multiplies and adds per output sample are required with this technique. Thus, the resonator method is roughly twice as efficient as the coupled form for generating pure sinewaves. If frequency modulation is desired, however, this advantage disappears.

3.2 Peak Amplitudes

Peak amplitude values are consistently ± 1 for the coupled form, regardless of word length precision. This doesn't always hold for the resonator method; peaks tend to be larger than 1 for small word lengths, and can even grow unstable. This phenomenon is worse for low frequencies than for high ones. However, with 16–18 fractional bits (to the right of the binary point), peak amplitudes resulting from the resonator method are indeed stable at 1.

3.3 Frequency Resolution

To assess frequency resolution (and also distortion), output waveforms were processed by Fourier transforms. A peak-finding algorithm was used to determine the fundamental frequency of the transform. It is not known how precise this algorithm is in determining frequencies, so results are stated informally.

Both methods have very accurate frequency resolution if at least 24 fractional bits are used, and even 20 are adequate in most cases. The coupled form slightly outperforms the resonator method in this regard, having highly accurate resolution for word lengths with more than 15 fractional bits. Fourteen fractional bits brings the accuracy to within 10 cents at 75 Hz.

3.4 Distortion

If less than 24 fractional bits are used with the resonator method, there is evident distortion in the output waveform. For low frequencies, this distortion is less than 10 dB below the signal when 14 fractional bits are used, and about 50 dB down when 16 fractional bits are used. At higher frequencies, distortion tends to be about 60 dB below the signal for fewer than 24 fractional bits. The modified coupled form produces stable, distortion-free sinewaves regardless of word length.

4. EXTENSIONS FOR FM

Neither of the two sine synthesis methods we have been discussing is easily modified to accommodate frequency modulation. However, the extensions required to realize this synthesis technique are not cost-prohibitive, though the coupled form offers more promise than the resonator method.

For proper frequency modulation, phase must be preserved. This means that if an output waveform is $\cos(n\omega)$, say, and the frequency is being modulated to a new value ω', the output must be interpreted as $\cos(m\omega')$, for some m such that $m\omega' = n\omega$. Thus, for the resonator method, $y(n-2)$, which would correspond to $\cos(n-1)\omega$ in our example, must be replaced by a value corresponding to $\cos(m-1)\omega'$. Since m doesn't correspond to any predetermined value, but rather is chosen so as to maintain phase continuity, it is not trivial to construct $\cos(m-1)\omega'$. Using a trigonometric identity, we have

$$\begin{aligned}\cos(m-1)\omega' &= \cos(m\omega')\cdot\cos(\omega') + \sin(m\omega')\cdot\sin(\omega')\\ &= \cos(n\omega)\cdot\cos(\omega') + \sin(n\omega)\cdot\sin(\omega')\end{aligned}$$

Thus, *two* resonators have to be computed in phase quadrature (one for $\cos(n\omega)$ and one for $\sin(n\omega)$), and both $\cos(\omega')$ and $\sin(\omega')$ have to be computed each sample (assuming general FM). Furthermore, four multiplies and two adds have to be performed each sample as well (a similar trigonometric identity applies to $\sin(m-1)\omega'$).

The same problem exists for the modified coupled form, but it appears to be more tractable. In this case, $\cos(n\omega - \varphi)$ must be replaced by $\cos(m\omega' - \varphi')$, and $\sin(n\omega - \varphi)$ with $\sin(m\omega' - \varphi')$. Trigonometric expansion results in

$$\begin{aligned}\cos(m\omega' - \varphi') &= \cos(m\omega')\cdot\cos(\varphi') + \sin(m\omega')\cdot\sin(\varphi')\\ &= \cos(n\omega)\cdot\frac{\varepsilon'}{2} + \sin(n\omega)\sin(\varphi')\\ &= \cos(n\omega)\cdot\sin(\frac{\omega'}{2}) + \sin(n\omega)\cos(\frac{\omega'}{2})\end{aligned}$$

and there is a similar equation involved for $\sin(m\omega' - \varphi')$. Two coupled forms must be computed in phase quadrature, but only one of the two equations in each pair needs to be computed (since $y(n+1)$ is immediately replaced by a newly computed value to accomplish the modulation).

In general, less computation will be involved in computing $\sin(\omega'/2)$ than in computing $\sin(\omega')$, and likewise for $\cos(\omega'/2)$ vs. $\cos(\omega')$. This ω' is the sum of a constant carrier frequency and a sinusoidally varying modulator; the sine and cosine functions can be approximated with series expansion. The error in this approximation can in general be higher than what could be tolerated in computing sinewaves by series expansion directly. It is postulated that a cubic for most circumstances — and a fifth-order polynomial in all cases — is sufficient to compute a new ϵ'. Simulations of FM synthesis with the coupled form are in progress, and results will be forthcoming shortly [Gordon (1985)].

5. CONCLUSIONS

The modified coupled first-order filter appears to be an attractive technique for synthesizing sine waves in VLSI. Only a few state variables are required to achieve absolutely stable sinusoids with accurate frequency resolution and no distortion. It also shows great promise in being extensible to frequency modulation synthesis. The modified coupled form compares favorably with the undamped digital resonator, especially for short word lengths and FM possibilities.

6. REFERENCES

Gordon, John W. (1985). Sine synthesis and frequency modulation in VLSI. *In Preparation.*

Lyon, Richard F. (1984). MSSP: A bit-serial multiprocessor for signal processing. *in VLSI Signal Processing,* (Workshop Proceedings); New York: IEEE Press.

Rabiner, L. R. & Gold, B. (1975). *Theory and Application of Digital Signal Processing*; p. 345. Englewood Cliffs, New Jersey: Prentice Hall.

Wawrzynek, J. C. & Mead, C. A. (1985). A VLSI architecture for sound synthesis. *in Denyer & Renshaw, eds., VLSI Signal Processing: A Bit-Serial Approach*; Reading: Addison Wesley.