Correspondence

Corrections to "A General Theory of Phase Noise in Electrical Oscillators"

Ali Hajimiri and Thomas H. Lee

The authors of the above paper[1] have found an error in (19) on p. 185. The factor of 8 in the denominator should be 4; therefore (19) should read

$$\mathcal{L}\{\Delta\omega\} = 10 \cdot \log\left(\frac{\frac{\overline{i_n^2}}{\Delta f}\sum_{n=0}^{\infty} c_n^2}{4q_{\max}^2 \Delta\omega^2}\right).$$

Noise power around the frequency $n\omega_0 + \Delta\omega$ causes two equal sidebands at $\omega_0 \pm \Delta\omega$. However, the noise power at $n\omega_0 - \Delta\omega$ has a similar effect as mentioned in the paper. Therefore, twice the power of noise at $n\omega_0 + \Delta\omega$ should be taken into account. This will also change the 4 in the denominator of (21) to 2 to read

$$\mathcal{L}\{\Delta\omega\} = 10 \cdot \log\left(\frac{\Gamma_{\rm rms}^2}{q_{\max}^2} \cdot \frac{\overline{i_n^2}/\Delta f}{2 \cdot \Delta\omega^2}\right).$$

Similarly, (24) must change, and its correct form is

$$\Delta\omega_{1/f^3} = \omega_{1/f} \cdot \left(\frac{c_0}{2\Gamma_{\rm rms}}\right)^2 \approx \omega_{1/f} \cdot \frac{1}{2}\left(\frac{c_0}{c_1}\right)^2.$$

This will result in the factor of 1/2 becoming redundant in (29), i.e.,

$$\mathcal{L}\{\Delta\omega\} = 10 \cdot \log\left(\frac{kT}{V_{\max}^2} \cdot \frac{1}{R_p \cdot (C\omega_0)^2} \cdot \left(\frac{\omega_0}{\Delta\omega}\right)^2\right).$$

However, note that the discussion following (29) is still valid.

The factor $c_0^2/2\Gamma_{\rm rms}^2$ should be changed to $(c_0/2\Gamma_{\rm rms})^2$ in the following instances:

1) p. 185, second column, last paragraph;
2) p. 190, second column, first paragraph;
3) p. 190, second column, second paragraph.

Nevertheless, the expression used to calculate the $\Gamma_{\rm rms}$ to predict phase noise of ring oscillators is based on a simulation that takes this effect into account automatically, and therefore the predictions are still valid. The authors regret any confusion this error may have caused.

Manuscript received February 27, 1998.

The authors are with the Center for Integrated Systems, Stanford University, Stanford, CA 94305-4070 USA.

Publisher Item Identifier S 0018-9200(98)03730-5.

[1] A. Hajimiri and T. H. Lee, *IEEE J. Solid-State Circuits*, vol. 33, pp. 179–194, Feb. 1998.

Comments on "A 64-Point Fourier Transform Chip for Video Motion Compensation Using Phase Correlation"[1]

Kevin J. McGee

***Abstract*— The fast Fourier transform (FFT) processor of the above paper,[1] contains many interesting and novel features. However, bit reversed input/output FFT algorithms, matrix transposers, and bit reversers have been noted in the literature. In addition, lower radix algorithms can be modified to be made computationally equivalent to higher radix algorithms. Many FFT ideas, including those of the above paper,[1] can also be applied to other important algorithms and architectures.**

I. Introduction

In the above paper,[1] the authors present a fast Fourier transform (FFT) processor that contains many interesting and novel features. The mathematics in the above paper,[1] describe a matrix computation where both time inputs and frequency outputs are in bit-reversed order. Bit-reversed input/output FFT algorithms, while not widely known, are not new, having been previously described in [3]. Fig. 1, for example, is a 16-point, radix-4, undecimated, bit reversed input/output, constant output geometry graph based on [3].

The algorithm[1] is also described as a decimation-in-time-and-frequency (DITF) type, but the architecture appears to be based on decimation-in-time (DIT). In the above paper,[1] Figs. 4 and 10 show a first calculation stage with unity twiddles before the butterfly and a second and third calculation stage with prebutterfly twiddles. Although the butterfly implementation of Fig. 5[1] may be unique, the use of prebutterfly twiddles in all three stages, along with unity twiddles in the first, would seem to indicate DIT. The architecture[1] is also a pipeline and contains many elements common to this type of processor, such as matrix transposers and bit reversers, as will be described below.

II. Matrix Transposers and Bit Reversers

Block serial/parallel or parallel/serial converters, sometimes called matrix transpose or corner turn buffers, are used in many systems. They perform a matrix transpose on data blocks by exchanging rows and columns. Fig. 2 (from [7]) shows, from upper left to lower right, the flow of data through a 4×4 shift-based transposer. The rotator lines show where data will be routed on the next clock cycle and the output is the transpose of the input. The switching action was noted in [7] and [8] and rotator designs can be found in [4], [7], and [8]. Although Fig. 6(b)[1] is also an 8×8 transposer, it is being used in a somewhat unusual way. By providing a complex (real and

Manuscript received January 31, 1997; revised March 5, 1998.

The author was with the Naval Undersea Warfare Center, Newport, RI 02841 USA. He is now at 33 Everett Street, Newport, RI 02840 USA.

Publisher Item Identifier S 0018-9200(98)03731-7.

[1] C. C. W. Hui, T. J. Ding, J. V. McCanny, and R. F. Woods, *IEEE J. Solid-State Circuits*, vol. 31, pp. 1751–1761, Nov. 1996.