A COMPARISON OF NOISE-SHAPING CLOCK GENERATORS FOR SWITCHED-CAPACITOR FILTERS *

Raymond W. Zeng† and Paul J. Hurst
Solid-State Circuits Research Laboratory
Department of Electrical and Computer Engineering
University of California, Davis, CA 95616

ABSTRACT

A programmable clock generator for switched-capacitor filters (SCFs) that uses noise shaping to generate a wide range of sampling frequencies with fine resolution was described in [1]. In this paper, alternative noise-shaping transfer functions are evaluated for the clock generator. Simulated and measured results are presented. Finally, a tone detector that uses a bandpass SCF driven by the clock generator with first-order noise shaping is described.

1. INTRODUCTION

The transfer function of a switched-capacitor filter (SCF) scales with its sampling frequency. A programmable clock generator that generates a wide range of sampling frequencies with fine resolution is described in [1]. The generated clock f_S has nonuniformly spaced sampling edges that align with the fast system clock f_C; such alignment helps to reduce coupling of digital switching noise into the SCF. The jitter in the sampling edges is concentrated at high frequencies by a noise-shaping loop, which is described below. When such a clock is used in a bandpass SCF, for example, its center frequency can be accurately set by adjusting the sampling frequency. Such adjustment can be used to precisely position a BPF in a tone-detection system or to correct slight SCF frequency-response errors caused by imperfect IC processing. One drawback of this approach is that the input to the filter must be bandlimited to well below f_S (e.g. f_{in}<$0.1f_S$ [1]), but this can often be achieved by using an imprecise lowpass filter before the critical SCF filter.

In [1], a sixth-order noise-shaping loop was used in the clock generator. Considering only the effect of sampling noise caused by the jitter in the clock f_S, the simulated signal-to-sampling-noise ratio (SSNR) at the filter output exceeds 70 dB for all f_{in}<$0.1f_S$ for two specific SCFs: a second-order BPF with Q=10 and f_O=$0.01f_S$ and a fifth-order LPF with $f_{-3\text{dB}}$=$0.04f_S$. However, measured results revealed that the output signal-to-noise ratio is limited by circuit noise sources (from the op amps and switched capacitors) rather than by sampling noise [1]. This result is important and means that lower order noise-shaping clock generators that give lower SSNR and require less IC area are worth investigating. This paper considers such clock generators and presents simulated and measured performance for them.

2. BACKGROUND

Fig. 1(a) shows a block diagram of a SCF driven by a nonuniform sampling clock f_S. The SCF samples the continuous-time input $s(t)$ and produces a sequence of output samples $y[n]$. A model for the SCF is shown in Fig. 1(b). Here, the input switch nonuniformly samples the input $s(t)$, producing a sequence of samples $x[n] = s(t_n)$, where t_n is the n^{th} sampling instant given by $t_n = nT + \tau[n]$. The desired sampling rate is $f_S = 1/T$, and $\tau[n]$ accounts for the jitter in the nonuniform sampling clock. If $|\tau|$ is small, τ is highpass and $s(t)$ is lowpass, $x[n]$ contains the input $s(t)$ sampled at the desired $1/T$ rate and highpass sampling noise due to the jitter [1]. The filter output samples $y[n]$ are found by convolving $x[n]$ with the impulse response of the SCF. When the SCF is a lowpass or bandpass filter, the transfer function $H(z)$ of the SCF attenuates the high-frequency sampling noise and gives a high SSNR at the filter output.

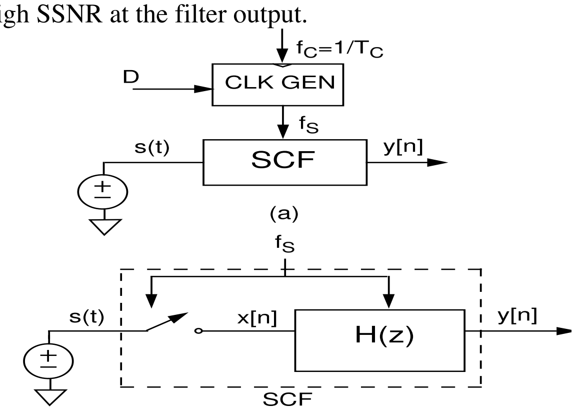

Fig. 1: (a) Block diagram of a SCF and nonuniform sampling clock generator. (b) A model for the SCF: a sampler followed by discrete-time convolution.

* This work was supported in part by a UC MICRO grant.

† R. Zeng is now with Intel Corp., Folsom, CA, 95630.

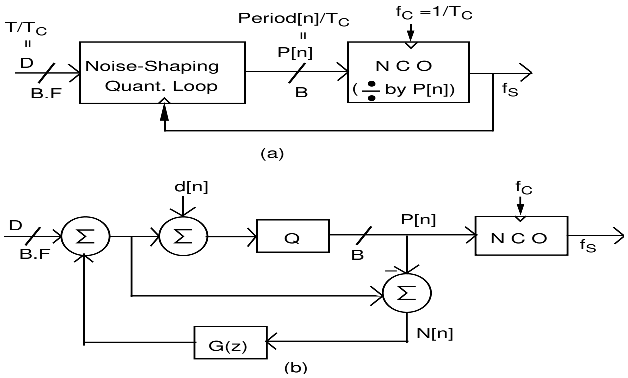

Fig. 2: (a) A block diagram of the clock generator. (b) The clock generator, showing the noise-shaping loop in more detail.

The clock generator is shown in Fig. 2(a) and consists of two major blocks: a noise-shaping quantization loop [2] and a number-controlled oscillator (NCO). The input to the noise-shaping loop is a constant D that equals the system clock frequency f_C divided by the desired sampling frequency f_S ($D = f_C/f_S = T/T_C$). Here D is a $B.F$-bit number; that is, D has B integer bits and F fractional bits. (In this paper, B=6 and F=6.) The output of the noise-shaping loop at time n is a B-bit integer $P[n]$. The NCO divides the fast clock f_C down by $P[n]$ which gives an instantaneous period equal to $P[n]T_C$ for the n^{th} sampling instant as shown in Fig. 3. A new P value is generated by the loop for each period of the clock f_S.

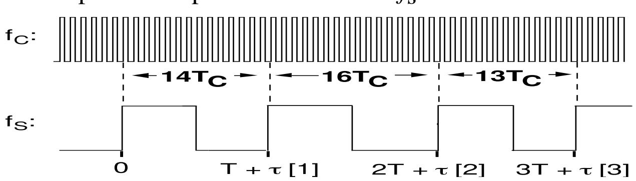

Fig. 3: Waveforms of the fast clock f_C and the nonuniform sampling clock f_S. Clock f_S is generated by dividing f_C by time-varying integer values (e.g., 14, 16, and 13 here). Sampling times $nT+\tau[n]$ are at the rising edges of f_S.

3. COMPARING DIFFERENT G(z)

In [1], a sixth-order $G(z)$ was used in the noise-shaping loop:

$$G_0(z) = 2z^{-1} - 0.5z^{-2} - 0.625z^{-3} - 0.125z^{-4} + 0.125z^{-5} + 0.125z^{-6}. \quad (1)$$

This transfer function gave a SSNR at the filter output that exceeds 70 dB for all $f_{in} < 0.1f_S$ for two specific SCFs: a second-order BPF with Q=10 and $f_O = 0.01f_S$ and a fifth-order LPF with $f_{-3\text{dB}} = 0.04f_S$. However, measured results revealed that the output SNR is limited to about 60 dB by circuit noise sources (from the op amps and switched capacitors) rather than by the sampling noise [1]. Therefore, lower order $G(z)$ that give lower SSNR and are simpler to implement are worth investigating.

The objectives when selecting a $G(z)$ are:

1- A sufficiently high simulated SSNR at the SCF output for input signals in a specified bandwidth (e.g. SSNR > 65 dB for $f_{in} < 0.1f_S$).

2- Low noise gain, which is the sum of the absolute value of the coefficients of $G(z)$ [3], to keep the peak-to-peak jitter in f_S small [1]. The peak-to-peak jitter determines the minimum half-clock period during which the op amp(s) must settle.

3- $G(z)$ should be low order and use mostly power-of-two coefficients to yield a simple, small IC implementation.

4- Highpass (or perhaps white) jitter.

To determine the spectrum of the jitter, the key transfer function is from the noise $N[n]$ in Fig. 2(b), which is assumed to be white, to the jitter $\tau[n]$ at the NCO output, and it is

$$H_T(z) = \frac{\tau(z)}{N(z)} = \frac{G(z)-1}{(1-z^{-1})f_C}. \quad (2)$$

This transfer function should be highpass (or possibly allpass) to give high SSNR in a low-frequency band.

Five different $G(z)$ that were investigated are listed in Table 1. Each $G(z)$ has at least one zero at z=1 to cancel the pole at z=1 in (2). (The pole at z=1 is due to the NCO.) Each G shapes the jitter spectrum differently, as determined by the $H_T(z)$ listed in the table. For $G_1(z)=z^{-1}$, $H_T(z)$ is a constant which means the jitter is white if $N[n]$ is white. $G_0(z)$ is the transfer function that was used in [1]. The other four $G(z)$ are lower order and are simpler to implement. Unfortunately, these $G(z)$ generate periodic $P[n]$, causing tones [4] to appear in the spectrum of $\tau[n]$. As a result, the sampling noise in $x[n]$ (see Fig. 1) consists of tones rather than shaped noise [5,6]. To eliminate these tones, a dither signal d is added to the input of the quantizer Q as shown in Fig. 2(b) [6]. This dither is a 0.4-bit pseudo-random number that is uniformly distributed from 0 to 1. The total noise $N[n]$ in Fig. 2(b) is $N[n] = d[n] + q[n]$, where $d[n]$ is the dither and $q[n]$ is quantization noise (the difference between the quantizer input and output).

$G(z)$	$H_T(z) \cdot f_C$	Noise Gain	% of CLB
$G_0 = 2z^{-1} - 0.5z^{-2} - 0.625z^{-3} - 0.125z^{-4} + 0.125z^{-5} + 0.125z^{-6}$	$(1-z^{-1})^2(1+0.5z^{-1}) \bullet (1+0.5z^{-1}+0.25z^{-2})$	3.5	98
$G_1 = z^{-1}$	-1	1	52
$G_2 = 2z^{-1} - z^{-2}$	$z^{-1} - 1$	3	62
$G_3 = 1.5z^{-1} - 0.5z^{-3}$	$(z^{-1}-1)(1+0.5z^{-1})$	2	79
$G_4 = 2z^{-1} - 2z^{-3} + z^{-4}$	$(1-z^{-1})^2(1+z^{-1})$	5	83

Table 1: Comparison of different $G(z)$

The noise gain for each G is listed in Table 1. The peak-to-peak jitter is determined by the noise gain. For G_0 with no dither signal added, the peak-to-peak jitter is $4T_c$ (4 is the noise gain of 3.5 rounded up to the next integer). Each of G_1-G_4 produces a peak-to-peak jitter equal to twice the noise gain times T_c because the added dither here doubles the peak-to-peak value of $N[n]$. Therefore, only G_1 gives a peak-to-peak jitter (= $2T_c$) that is less than the peak-to-peak jitter with G_0.

4. SIMULATED AND MEASURED RESULTS

For the following simulated and measured results, the test conditions were: f_c = 2 MHz, D = 20.25 (which gives f_s = 98.77 kHz), f_{in} = 1 kHz and V_{in} = 1.5 $V_{p\text{-}p}$, and the second-order BPF and fifth-order LPF described above are used. (The SSNR is independent of amplitude, so the choice of amplitude is not important here [1]. Additional information on the BPF and LPF is in [1].). Dither is added for G_1-G_4. Figures 4 and 5 show the spectra of the input samples $x[n]$ from simulation of the model of Fig. 1(b) using G_1 and G_4, respectively, in the noise-shaping loop. Notice that the noise floors are shaped differently and in agreement with the corresponding H_T's in Table 1. The simulated spectra of the LPF output, $|Y(\omega)|$, are shown in Figures 6 and 7 for G_1 and G_4. Both spectra show the effect of the LPF's high-frequency attenuation on the noise floor. The passband noise floor for G_1 is much higher than for G_4, and the simulated SSNR for G_4 is 87.8 dB, which is much higher than the SSNR of 68.4 dB for G_1. The simulated SSNRs at the LPF and BPF outputs for all the G's are given in Table 2.

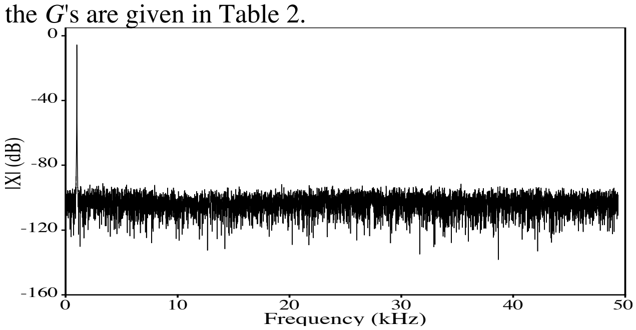

Fig. 4: Simulated spectrum of the samples $x[n]$ in Fig. 1(b) with G_1 in the noise-shaping clock generator.

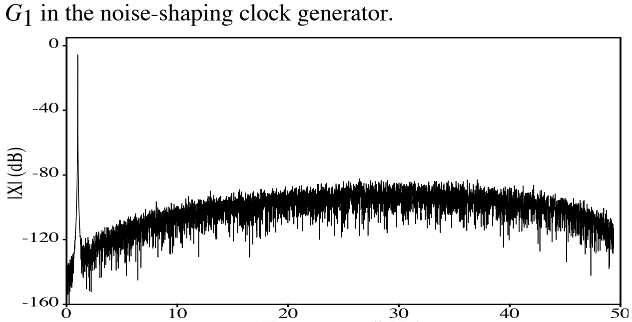

Fig. 5: Simulated spectrum of the samples $x[n]$ in Fig. 1(b) with G_4 in the noise-shaping clock generator.

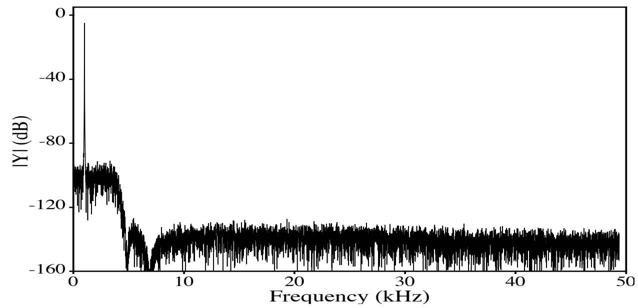

Fig. 6: Simulated spectrum of the LPF output with G_1 in the noise-shaping loop.

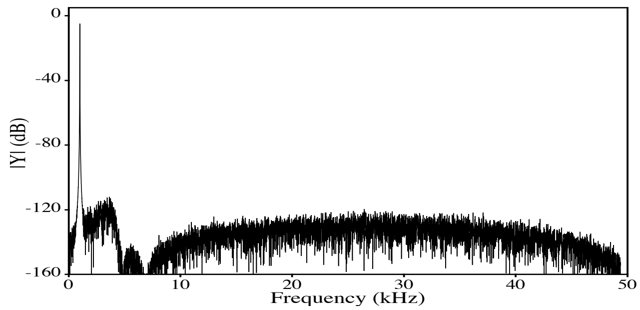

Fig. 7: Simulated spectrum of the LPF output with G_4 in the noise-shaping loop.

G(z)	LPF		BPF	
	Simulated SSNR(dB)	Measured SNR(dB)	Simulated SSNR(dB)	Measured SNR(dB)
G_0	84.2	61.9	86.6	63.0
G_1	68.4	61.7	82.9	63.8
G_2	84.2	60.4	99.9	61.7
G_3	80.1	60.5	88.1	62.6
G_4	87.8	58.1	98.7	59.2

Table 2: SNR at filter outputs for different $G(z)$

The clock generator of Fig. 2(b) was implemented using a Xilinx XC4005 chip so that the clock generator could be tested with different $G(z)$. As a measure of complexity, the percentage of configurable logic blocks (CLBs) used in the Xilinx chip to implement the different clock generators is given in the last column of Table 1. G_1-G_4 are built with a dither generator, and the complexity of the dither generator is included in the CLB counts. The clock generator using G_0 was also programmed (without a dither generator) for comparison. Measured results were taken using the different clock generators with the second-order BPF and fifth-order LPF that were integrated in a 2-μm CMOS process [1]. Measurements confirmed that the clock generators using these five different $G(z)$ produced a wide range of sampling frequencies with the same resolution. Therefore, all the clock generators provided the same control over the frequency response of the SCFs.

Figures 8 and 9 show the measured spectra of LPF outputs for G_1 and G_4, respectively. These spectra were found by sampling the SCF output with a 14-bit ADC that is clocked by f_S and then taking a FFT of the samples. The noise floor is higher here at both high and low frequencies than simulated, which means that circuit noise is much larger than sampling noise. Comparing Figs. 8 and 9, the noise floor for G_4 in the filter stopband is somewhat higher than for G_1, whereas the passband noise floors are about the same. As a result, the measured signal-to-noise ratio (SNR), which includes the effect of sampling and circuit noise but excludes distortion, is larger for G_1 than for G_4. (See Table 2.) This result is surprising since the simulated SSNR for G_1 is smaller than the SSNR for G_4. The reason for this discrepancy between the relative SSNRs and SNRs is uncertain, but it could be either that the SCFs do not provide as much high-frequency attenuation of the noise as predicted by the model, or there is a source of noise in a nonuniformly clocked SCF that we do not yet understand. (As evidence of this, measured SNRs with a uniform sampling clock (D=20) are 66.3dB for the LPF and 67.7dB for the BPF. These values are noticeably larger than the measured SNRs in Table 2.)

The measured SNRs at the BPF and LPF outputs for all the G's are given in Table 2. Comparison of these SNRs for both filters reveals that G_1 has the highest SNR and G_4 has the lowest among G_1-G_4. Therefore, considering SNR, peak-to-peak jitter, and complexity, $G_1(z)$ is the best choice for these SCFs.

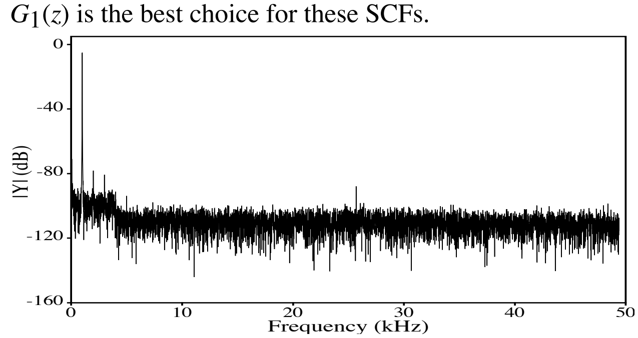

Fig. 8: Measured spectrum at the output of the SC LPF with G_1 in the noise-shaping clock generator.

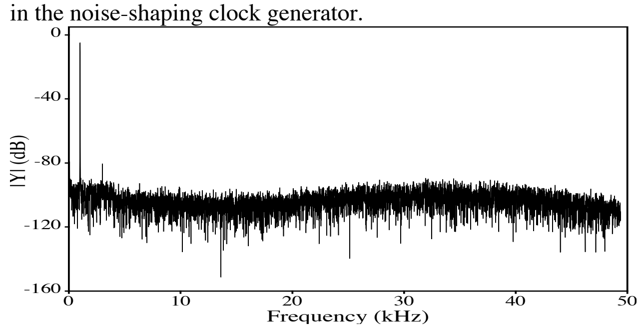

Fig. 9: Measured spectrum at the output of the SC LPF with G_4 in the noise-shaping clock generator.

5. TONE DETECTOR EXAMPLE

One possible application of this clock generator is in a system that detects signaling tones, such as those used in a dual-tone multifrequency (DTMF) receiver. For DTMF, tones at 8 different frequencies (697 Hz, 770 Hz, 852 Hz, 941 Hz, 1209 Hz, 1336 Hz, 1477 Hz, and 1633 Hz) with amplitudes between –2 dBm and –32 dBm must be detected [7]. A block diagram of a tone detector that uses a programmable SCF is shown in Fig. 10. An imprecise second-order RC LPF with $f_{-3dB}<0.1f_S$ ($f_{-3dB}\approx5$ kHz) is used as a bandlimiting filter. After the RC LPF, the signal passes through a high-Q programmable SC BPF (Q=10, $f_O=0.01f_S$). The center frequency of this BPF is programmed by the noise-shaping clock generator that uses G_1. This BPF passes a narrow band centered at the tone frequency of interest. The filter output is then digitized and processed digitally to determine if a sine wave with the correct frequency and sufficient amplitude is present. With this system, different tones can be detected by changing D which changes the sampling frequency f_S and the center frequency of the BPF. With D=6.6 bits and f_C=2 MHz, the center frequency of the BPF can be set to within 0.06% of any of the 8 DTMF tone frequencies. With simple amplitude and frequency qualification in the DSP [7], this system detects tones with a frequency that is within $\pm$2.3% of the desired tone frequency with amplitudes between 10 dBV and –40 dBV.

6. CONCLUSION

Noise-shaping clock generators using simple $G(z)$ and dither generate a wide range of sampling frequencies for SCFs. A simple first-order clock generator achieves acceptable performance for many filtering applications.

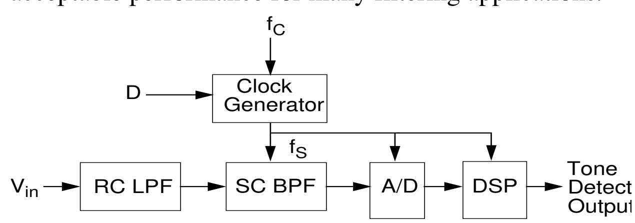

Fig. 10: DTMF tone detector using a SC BPF and a noise-shaping clock generator. (DSP = Xilinx XC4005)

REFERENCES

[1] P. J. Hurst and B. C. Rothenberg, "A programmable clock generator that uses noise shaping and its application in switched-capacitor filters," *IEEE J. Solid-State Circuits*, vol. 30, no. 4, pp. 403-411, April 1995.

[2] S. K. Tewksbury and R. W. Hallock, "Oversampled, linear-predictive and noise-shaping coders of order N>1," *IEEE Trans. on CAS,* vol. 25, no. 7, pp. 436-447, July 1978.

[3] L. R. Carley and J. Kenney, "A 16-bit 4th order noise-shaping D/A converter," *CICC*, pp. 21.7.1-21.7.4, 1988.

[4] J. C. Candy, O. J. Benjamin, "The structure of quantization noise from sigma-delta modulation," *IEEE Trans. on Comm.*, vol. 29, pp. 1316-1323, Sept. 1981.

[5] W. Chou, R. M. Gray, "Dithering and its effects on sigma-delta and multistage sigma-delta modulation," *IEEE Trans. on Inform. Theory*, vol. 37, no. 3, pp. 500-513, May 1991.

[6] S. R. Norsworthy, "Optimal Nonrecursive noise shaping filters for oversampling data converters part1: theory," *ISCAS*, pp. 1353-1356, 1993.

[7] B. J. White, G. M. Jacobs, and G. F. Landsburg, "A monolithic dual tone multifrequency receiver," *IEEE J. Solid-State Circuits*, vol. 14, no. 6, pp. 991-997, Dec. 1979.