CLOCK JITTER ESTIMATION IN NOISE

Zaid J. Towfic and Ali H. Sayed
University of California - Los Angeles
Department of Electrical Engineering
University of California, Los Angeles, CA 90095
Email: {ztowfic,sayed}@ee.ucla.edu

***Abstract*—Clock timing jitter refers to random perturbations in the sampling time in analog-to-digital converters (ADCs). The perturbations are caused by circuit imperfections in the sampling clock. This paper analyzes the effect of sampling clock jitter on the acquired samples in the midst of quantization noise and random Gaussian noise. The paper proposes a method for estimating the jitter for cognitive radio architectures at high sampling rates. The paper also examines the fixed-point implementation of the algorithm and its theoretical performance.**

***Index Terms*—Clock jitter, analog-to-digital conversion (ADC), noise, quantization**

I. INTRODUCTION

An analog-to-digital converter (ADC) is a basic building block of modern communication systems. Certain applications of modern radios, such as cognitive radios and UWB radios, may require ADCs to operate at high sampling rates due to the use of wide frequency bandwidths. At high rates, signal distortion is introduced by clock jitter. The jitter causes the ADC to sample the input signal along a non-uniform sampling grid and introduces distortion that limits the signal fidelity and degrades the signal-to-noise ratio (SNR) [1]. In cognitive radio applications, the jitter degrades the spectrum sensing performance [2].

In general, clock jitter can be reduced at the analog circuitry level by improving the PLL design. Digital recovery of the jitter, however, allows the circuit constraints to be relaxed in favor of slightly greater load on the signal processor. It is known that clock jitter is correlated and tends to have a spectrum that falls as f^{-2} [3], [4]. This fact enables digital recovery methods that exploit the spectral properties of clock jitter, such as the training tone injection method [1]. In this work, we first examine the effect of sampling clock jitter on the SNR of the sampled signal when the signal is sampled with quantization and random noise. We then propose a method to estimate the jitter for cognitive radio architectures where the signals of interest lie in the low-frequency range. The method is implemented in fixed point and the performance is examined for initial jitter RMS of 1% of the sampling period.

II. EFFECT OF CLOCK JITTER

The effect of sampling jitter on a sampled ideal low-pass signal was studied in [1]. The analysis did not include the effect of quantization noise and other noise sources such as random noise at the ADC. In order to expand the analysis, let us examine the SNR performance of a b-bit ADC (including the sign bit) with jitter, quantization noise, and random noise, when it is fed with an ideal low-pass signal $d(t)$ with baseband bandwidth of B Hz. The sampled signal can be expressed as:

$$\begin{aligned} r(n) &= d(t+e(t))|_{t=nT_s} + q(n) + v(n) \\ &\approx d(n) + e(n)\dot{d}(n) + q(n) + v(n) \end{aligned}$$

This work was supported in part by DARPA contract N66001-09-1-2029.

where

$$\dot{d}(n) \triangleq \dot{d}(t)\Big|_{t=nT_s}$$

$d(n)$ is the desired signal, $q(n)$ is the quantization noise for the n-th sample, $v(n)$ is zero-mean white Gaussian random noise with variance σ_v^2, and $e(n)$ is the jitter noise at the n-th sample. In this paper, we assume that the jitter has the following properties:

1) $e(n)$ is Gaussian distributed with variance σ_e^2.
2) $e(n)$ is correlated with passband bandwidth of f_e Hz.
3) $e(n)$ is independent of $d(n)$ and other noise sources.

It is well known that the quantization noise power is given by [5]

$$\begin{aligned} E\left[q(n)^2\right] &= \frac{\left(\frac{R}{2^b}\right)^2}{12} \\ &= \frac{1}{3\cdot 4^b} \quad (\text{Assuming } R=2) \end{aligned}$$

where R is the range of the ADC and the jitter $q(n)$ is assumed to be uniformly distributed with zero-mean and is white.

A. Wideband Signal Model

In [1], [6], the variance of $e(n)\dot{d}(n)$ for the ideal lowpass signal was approximated as:

$$E\left[\left(e(n)\dot{d}(n)\right)^2\right] \approx \frac{1}{3}\left(2\pi B\sigma_d\sigma_e\right)^2$$

so that the signal-to-noise ratio at the output of the ADC can be expressed as:

$$\text{SNR} = \frac{\sigma_d^2}{\frac{1}{3}\left(2\pi B\sigma_d\sigma_e\right)^2 + \frac{1}{3\cdot 4^b} + \sigma_v^2} \tag{1}$$

Throughout the analysis, we set σ_e^2 as:

$$\sigma_e^2 = (\alpha T_s)^2$$

This allows the variance of the jitter to be defined in terms of the sampling period. The above implies that the jitter noise dominates the SNR expression when

$$\frac{1}{3}\left(2\pi B\sigma_d\sigma_e\right)^2 \gg \frac{1}{3\cdot 4^b} + \sigma_v^2$$

or, equivelently,

$$\sigma_e \gg \frac{\sqrt{4^{-b}+3\sigma_v^2}}{2\pi B\sigma_d}$$

B. Sinusoidal Signal Models

In the special case when

$$d(t) = A\cdot\cos(2\pi f_d t + \theta_s)$$

we get

$$E\left[\left(e(n)\dot{d}(n)\right)^2\right] \approx \frac{1}{2}\left(2\pi f_d A\sigma_e\right)^2$$

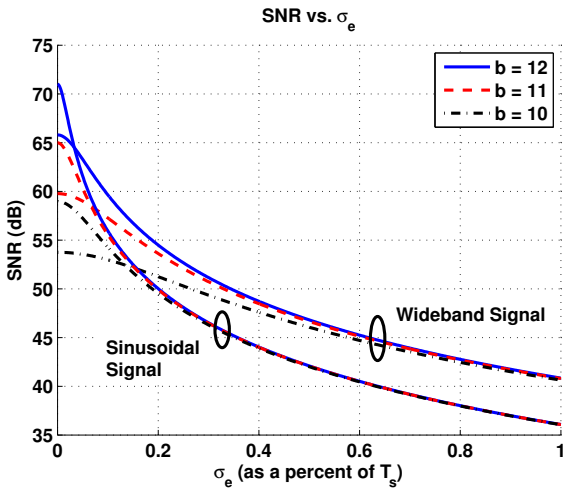

Fig. 1. SNR vs. σ_e for $f_s = 1$GHz, $B = 250$MHz, and $\sigma_d \approx 0.388$

and the SNR equation (1) becomes

$$\mathrm{SNR} = \frac{\frac{A^2}{2}}{\frac{1}{2}\left(2\pi f_d A \sigma_e\right)^2 + \frac{1}{3\cdot 4^b} + \sigma_v^2} \tag{2}$$

In order to illustrate the SNR relationship derived for both signal models, the ADC range is assumed to be $[-1, 1]$ and the signal $d(n)$ is assumed to be Gaussian with baseband bandwidth of 250MHz in the wideband case (1) and a sinusoid at frequency 250MHz and amplitude $A = 1$ in the sinusoidal case (2). The sampling frequency is fixed at $f_s = 1$GHz. The signal $d(n)$ is assumed to overflow the ADC at most 1% of the time in the wideband signal case, and hence $\sigma_d = \frac{1}{Q^{-1}(0.005)}$ where $Q^{-1}(\cdot)$ is the inverse Q function defined by

$$Q^{-1}(z) \triangleq \frac{1}{\sqrt{2\pi}} \int_z^\infty \exp\left(-\frac{t}{2}\right) dt$$

Figure 1 illustrates this relationship for $f_s = 1$GHz, $B = 250$MHz, $A = 1$, and $\sigma_d \approx 0.388$.

III. ESTIMATION OF CLOCK JITTER

A. Superheterodyne Receiver

In a superheterodyne receiver, the desired signal is a passband signal and is located at high frequencies. If a low-frequency tone is injected into the system, it is possible to examine this tone after sampling and to estimate the jitter $e(n)$. This technique was proposed in [1]. Here, we examine the optimal estimator for the jitter $e(n)$ in the presence of quantization noise, $q(n)$, and Gaussian noise, $v(n)$. For this purpose, we examine the performance of the algorithm presented in [1], which recovers the jitter $e(n)$ as – see Figure 2:

$$\begin{aligned}\hat{e}(n) &= \mathrm{LPF}\left\{\frac{r(n)\sin\left(2\pi f_w n T_s + \theta_w\right)}{-\pi f_w \delta}\right\} \\ &= e(n) + \nu(n)\end{aligned}$$

where

$$\nu(n) \approx \mathrm{LPF}\left\{\frac{q(n)\sin\left(2\pi f_w n T_s + \theta_w\right)}{-\pi f_w \delta} + \frac{v(n)\sin\left(2\pi f_w n T_s + \theta_w\right)}{-\pi f_w \delta}\right\}$$

is approximately Gaussian if a FIR filter of sufficient length is used (LPF denotes a low-pass filtering operation). If no further filtering

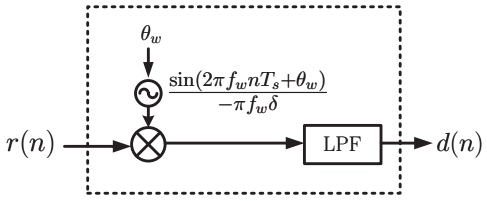

Fig. 2. Block diagram for jitter estimation for Superheterodyne receiver

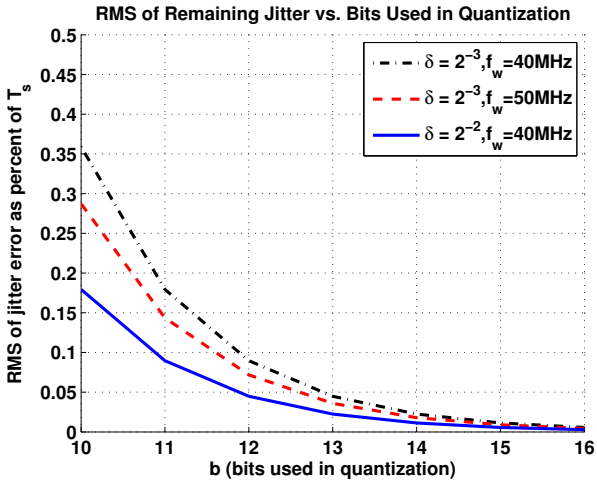

Fig. 3. Theoretical RMS of remaining jitter assuming $f_s = 1$GHz and $f_e = 5$MHz.

takes place, then the RMS of the remaining jitter will be dictated by the power in $\nu(n)$. More specifically,

$$\begin{aligned}\sigma_\nu \cdot f_s \cdot 100 &= \sqrt{E\left[\nu(n)^2\right]} \cdot f_s \cdot 100 \\ &\approx \sqrt{\frac{\left(\frac{2f_e}{f_s}\right)}{2\left(-\pi f_w \delta\right)^2} \cdot \left(E\left[q(n)^2\right] + E\left[v(n)^2\right]\right)} \cdot f_s \cdot 100 \\ &= \left(\frac{1}{\pi f_w \delta}\right) \cdot \sqrt{f_e f_s} \cdot \sqrt{\frac{1}{3\cdot 4^b} + \sigma_v^2} \cdot 100\end{aligned}$$

where the factor $\frac{2f_e}{f_s}$ scales the power of the white processes $q(n)$ and $v(n)$ after being filtered by an ideal low-pass filter with baseband cut-off frequency at f_e. In fact, since the noise is approximately Gaussian, the filter that minimizes the remaining jitter is known to be linear [7]. This is left to Section III-C. Figure 3 shows the performance of the estimation for $f_s = 1$GHz, $f_e = 5$MHz and for varying values of δ and f_w. The power of the Gaussian random noise is assumed to be the same as that of the quantization noise, i.e., $\sigma_v^2 = \frac{1}{3\cdot 4^b}$. The number of bits used in the quantization (and, hence, the power of the noise) is swept from $b = 10$ to $b = 16$. Note that in this simulation, δ represents the amplitude of the injected tone (which limits the dynamic range of the received data).

B. High Frequency Signal Injection

As indicated before, a method for estimating the jitter of a high frequency tone was introduced. In this paper, we propose an algorithm that achieves a lower error variance. The same tone injection method will be used as in [1]. Consider the injected tones before the ADC:

$$\begin{aligned}p(t) &= 2\delta w(t) \cdot y(t) \\ &= 2\delta \cos(2\pi f_w t + \theta_w) \cdot \cos(2\pi f_y (t + \tau(t)) + \theta_y)\end{aligned}$$

$$= \delta\cos(2\pi(f_y - f_w)t + 2\pi f_y\tau(t) + \theta_y - \theta_w) +$$
$$\delta\cos(2\pi(f_y + f_w)t + 2\pi f_y\tau(t) + \theta_y + \theta_w)$$

where $\tau(t)$ is the jitter of the high frequency oscillator resonating at frequency f_y. After the ADC, the received samples become:

$$\begin{aligned}
r(n) &= (1-2\delta)\left(d(n) + \dot{d}(n)e(n)\right) + q(n) + v(n) + \\
&\quad \delta\cos(2\pi(f_y - f_w)(nT_s + e(n)) + 2\pi f_y\tau(n) + \theta_y - \theta_w) + \\
&\quad \delta\cos(2\pi(f_y + f_w)(nT_s + e(n)) + 2\pi f_y\tau(n) + \theta_y + \theta_w) \\
&= (1-2\delta)\tilde{d}(n) + q(n) + v(n) + \\
&\quad \delta\cos(2\pi(f_y - f_w)(nT_s + e(n) + \frac{f_y\tau(n)}{f_y - f_w}) + \theta_y - \theta_w) + \\
&\quad \delta\cos(2\pi(f_y + f_w)(nT_s + e(n) + \frac{f_y\tau(n)}{f_y + f_w}) + \theta_y + \theta_w)
\end{aligned}$$

where $e(n)$ is the jitter of the ADC. Downconversion of the two tones at $f_y - f_w$ and $f_y + f_w$ can be verified to yield

$$\begin{aligned}
x(n) &= \text{LPF}\left\{r(n)\sin(2\pi(f_y - f_w)nT_s + \theta_y - \theta_w)\right\} \\
&= \frac{-\delta}{2}\sin\left(2\pi(f_y - f_w)\left(e(n) + \frac{f_y}{f_y - f_w}\tau(n)\right)\right) + \\
&\quad \text{LPF}\left\{q(n)\sin\left(2\pi(f_y - f_w)nT_s + \theta_y - \theta_w\right)\right\} + \\
&\quad \text{LPF}\left\{v(n)\sin\left(2\pi(f_y - f_w)nT_s + \theta_y - \theta_w\right)\right\} \\
&\approx -\delta\pi(f_y - f_w)\left(e(n) + \frac{f_y}{f_y - f_w}\tau(n)\right) + q_1(n) + v_1(n) \\
&= -\delta\pi(f_y - f_w)g(n) + \nu_1(n)
\end{aligned}$$

and

$$\begin{aligned}
z(n) &= \text{LPF}\left\{r(n)\sin(2\pi(f_y + f_w)nT_s + \theta_y + \theta_w)\right\} \\
&= \frac{-\delta}{2}\sin\left(2\pi(f_y + f_w)\left(e(n) + \frac{f_y}{f_y + f_w}\tau(n)\right)\right) + \\
&\quad \text{LPF}\left\{q(n)\sin\left(2\pi(f_y + f_w)nT_s + \theta_y + \theta_w\right)\right\} + \\
&\quad \text{LPF}\left\{v(n)\sin\left(2\pi(f_y + f_w)nT_s + \theta_y + \theta_w\right)\right\} \\
&\approx -\delta\pi(f_y + f_w)\left(e(n) + \frac{f_y}{f_y + f_w}\tau(n)\right) + q_2(n) + v_2(n) \\
&= -\delta\pi(f_y + f_w)h(n) + \nu_2(n)
\end{aligned}$$

where the last approximation assumes that the argument of $\sin(\cdot)$ is small and

$$\begin{aligned}
g(n) &= e(n) + \frac{f_y}{f_y - f_w}\tau(n) \\
h(n) &= e(n) + \frac{f_y}{f_y + f_w}\tau(n) \\
q_1(n) &= \text{LPF}\left\{q(n)\sin\left(2\pi(f_y - f_w)nT_s + \theta_y - \theta_w\right)\right\} \\
q_2(n) &= \text{LPF}\left\{q(n)\sin\left(2\pi(f_y + f_w)nT_s + \theta_y + \theta_w\right)\right\} \\
v_1(n) &= \text{LPF}\left\{v(n)\sin\left(2\pi(f_y - f_w)nT_s + \theta_y - \theta_w\right)\right\} \\
v_2(n) &= \text{LPF}\left\{v(n)\sin\left(2\pi(f_y + f_w)nT_s + \theta_y + \theta_w\right)\right\} \\
\nu_1(n) &= q_1(n) + v_1(n) \\
\nu_2(n) &= q_2(n) + v_2(n)
\end{aligned}$$

Finally,

$$\begin{aligned}
\hat{e}(n) &= \frac{x(n) - z(n)}{2\pi f_w\delta} \\
&\approx e(n) + \frac{1}{2\pi f_w\delta}\left(\nu_1(n) - \nu_2(n)\right) \\
&= e(n) + \nu(n)
\end{aligned}$$

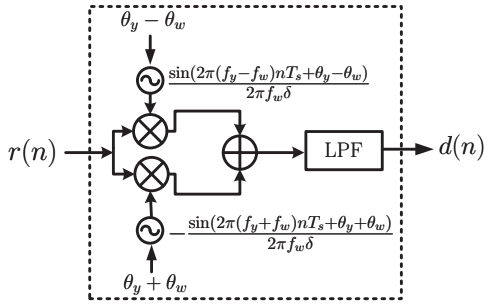

Fig. 4. Block diagram representation of the recovery algorithm in (3).

Note that the LPF $\{\cdot\}$ operation is a linear operation and hence the algorithm can be simplified to

$$\boxed{\hat{e}(n) = \text{LPF}\left\{\frac{r(n)\sin(2\pi(f_y - f_w)nT_s + \theta_y - \theta_w)}{2\pi f_w\delta} - \frac{r(n)\sin(2\pi(f_y + f_w)nT_s + \theta_y + \theta_w)}{2\pi f_w\delta}\right\}} \quad (3)$$

The algorithm is illustrated in Figure 4. If a FIR low-pass filter of sufficient length is used, then it is reasonable to expect that $\nu(n)$ will be Gaussian and an optimal linear estimation filter can be derived – see Section III-C. If no further filtering takes place, the error in the estimation will be determined by the power in $\nu(n)$. Specifically, if an ideal low-pass filter is used in (3), then:

$$\begin{aligned}
E\left[\nu(n)^2\right] &= \left(\frac{1}{2\pi f_w\delta}\right)^2\left(E\left[\nu_1(n)^2\right] + E\left[\nu_2(n)^2\right]\right) \\
&\approx \left(\frac{1}{2\pi f_w\delta}\right)^2 \cdot \left(E\left[q_1(n)^2\right] + E\left[v_1(n)^2\right] + \right. \\
&\qquad \left. E\left[q_2(n)^2\right] + E\left[v_2(n)^2\right]\right)
\end{aligned}$$

Since $E\left[q_1(n)^2\right] = E\left[q_2(n)^2\right]$ and $E\left[v_1(n)^2\right] = E\left[v_2(n)^2\right]$, we have

$$\begin{aligned}
E\left[\nu(n)^2\right] &\approx \left(\frac{1}{2\pi f_w\delta}\right)^2 \cdot 2 \cdot \left(E\left[q_1(n)^2\right] + E\left[v_1(n)^2\right]\right) \\
&\approx \left(\frac{1}{2\pi f_w\delta}\right)^2 \cdot \left(\frac{2f_e}{f_s}\right) \cdot \left(E\left[q(n)^2\right] + E\left[v(n)^2\right]\right) \\
&= \left(\frac{1}{2\pi f_w\delta}\right)^2 \cdot \left(\frac{2f_e}{f_s}\right) \cdot \left(\frac{1}{3\cdot 4^b} + \sigma_v^2\right)
\end{aligned}$$

where the factor $\frac{2f_e}{f_s}$ scales the power of the white processes $q(n)$ and $v(n)$ after being filtered by an ideal low-pass filter with baseband cut-off frequency of f_e. It can be shown that the method proposed in [1] achieves a noise variance of

$$E\left[\nu_{[1]}(n)^2\right] \approx \left(\frac{1}{\pi f_w\delta}\right)^2 \cdot \left(\frac{2f_e}{f_s}\right) \cdot \left(\frac{1}{3\cdot 4^b} + \sigma_v^2\right)$$

and

$$E\left[\nu_{[1]}(n)^2\right] \approx 4E\left[\nu(n)^2\right]$$

Hence, the RMS of the remaining jitter as a percent of the sampling time after perfect compensation will be

$$\begin{aligned}
\text{RMS}\left\{\tilde{e}(n)\right\} &= \sqrt{E\left[(\hat{e}(n) - e(n))^2\right]} \cdot f_s \cdot 100 \\
&\approx \sqrt{\left(\frac{1}{2\pi f_w\delta}\right)^2 \cdot \left(\frac{2f_e}{f_s}\right) \cdot \left(\frac{1}{3\cdot 4^b} + \sigma_v^2\right)} \cdot f_s \cdot 100
\end{aligned}$$

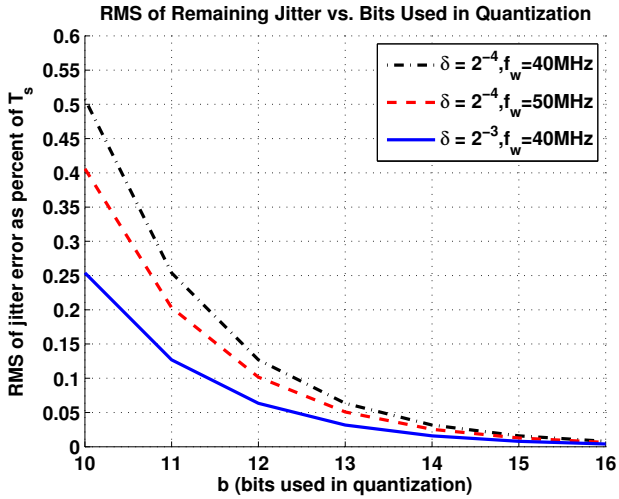

Fig. 5. Theoretical RMS of remaining jitter assuming $f_s = 1\text{GHz}$ and $f_e = 5\text{MHz}$

$$\approx \left(\frac{1}{2\pi f_w \delta}\right) \cdot \sqrt{2 f_e f_s} \cdot \sqrt{\frac{1}{3 \cdot 4^b} + \sigma_v^2} \cdot 100$$

Figure 5 shows the performance of the estimation for $f_s = 1\text{GHz}$, $f_e = 5\text{MHz}$ and for varying values of δ and f_w. The power of the Gaussian random noise is assumed to be the same as that of the quantization noise, i.e., $\sigma_v^2 = \frac{1}{3 \cdot 4^b}$. The number of bits used in the quantization (and, hence, the power of the noise) is swept from $b = 10$ to $b = 16$. It is important to note that δ in this simulation represents half the amplitude of the injected tones as each tone has amplitude δ and hence the maximum amplitude of the training tones is 2δ. This would mean that a fair comparison between the high-frequency injection algorithm and the low-frequency injection algorithm must take the value of δ into account. In fact, this implies that the remaining jitter RMS value of the high frequency recovery is $\sqrt{2}$ times the RMS value of the low-frequency recovery for the same values of f_e,f_s, f_w, and equivalent values of δ.

C. Optimum Estimator

In the previous two sections, an estimate of the jitter was found by extracting it from the training tones injected into either the low-frequency region or the high-frequency region. The problem can be cast into the estimation framework as follows. Let

$$\hat{e} = e + \nu$$

where $\hat{e}$, e, and ν are vectors of length N containing $[d(0), d(1), \cdots, d(N-1)]^\top$, $[e(0), e(1), \cdots, e(N-1)]^\top$, and $[\nu(0), \nu(1), \cdots, \nu(N-1)]^\top$ respectively. If a FIR low-pass filter of sufficient length was used in the computation of $x(n), z(n)$, then the central-limit theorem suggests that $\nu(n)$ will be approximately Gaussian. This implies that the optimal mean-square-error estimator of e is the linear estimator [7]:

$$\hat{e}_{\text{opt}} = R_{ed} R_d^{-1} \hat{e} = R_e \left(R_e + R_\nu\right)^{-1} \hat{e}$$

where the covariance matrix R_d depends on the LPF filter used. The MMSE matrix is then:

$$\text{MMSE} = \left(R_e^{-1} + R_\nu^{-1}\right)^{-1}$$

The m.m.s.e. in estimating $\mathbf{e}$ from $\mathbf{d}$ is $\frac{1}{N}\text{Tr}\left[\text{MMSE}\right]$, where $\text{Tr}[\cdot]$ is the trace operator. The spectrum of the jitter has been found to be Lorentzian in shape which falls as f^{-2} [3], [4], [8] while the noise spectrum is shaped by the low-pass filter. This implies that further filtering will improve the MSE of the estimation procedure.

IV. IMPLEMENTATION

The algorithm presented in (3) has been implemented in fixed point with Mathworks Simulink and the Synopsys SynDSP toolbox. One bottleneck that drives the performance of the algorithm is the low-pass filter. The low-pass filter in the Simulink implementation was designed as an "Accumulate-downsample-upsample" filter which accumulates 50 samples, downsamples to the sum of 50 samples each and finally upsamples using a linear interpolator. This design is not a direct FIR filter, however, its architecture stills requires summing 50 samples of the noise which works to satisfy the central limit theorem once again. The low-pass filter block is illustrated in Figure 6.

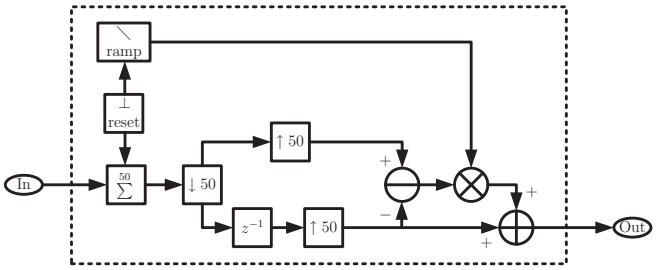

Fig. 6. Lowpass filter structure implemented in fixed-point simulation.

The design was implemented using Synopsys Premier on a Xilinx Virtex5 board (XC5VLX20T). The design requires a total of 544 Look-up-tables. Two tones located at $f_w = 40\text{MHz}$ and $f_y = 410\text{MHz}$ are injected into the system. The Gaussian processes $e(n)$ and the jitter $\tau(n)$ are both assumed to have the same statistical characteristics, namely 5MHz bandwidth and standard deviation $\frac{1}{100}T_s$. δ is set at $\delta = 2^{-4}$, $b = 10$, and $T_s = 1\text{ns}$. Simulation shows that the fixed point model is able to reduce the jitter from 1% to 0.5% of T_s. With an improved low-pass filter, the performance closely tracks that of Figure 5.

REFERENCES

[1] Z. Towfic, S.-K. Ting, and A. H. Sayed, "Sampling clock jitter estimation and compensation in ADC circuits," in *Proc. 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2010, pp. 829 –832.

[2] M. Oner, "On the effect of random sampling jitter on cyclostationarity based spectrum sensing algorithms for cognitive radio," in *Proc. IEEE 69th Vehicular Technology Conference, Barcelona, Spain*, Apr. 2009, pp. 1 –5.

[3] M. Löhning and G. Fettweis, "The effects of aperture jitter and clock jitter in wideband ADCs," *Computer Standards & Interfaces*, vol. 29, no. 1, pp. 11–18, 2007.

[4] B. Razavi, *RF Microelectronics*. Prentice Hall, NJ, 1998.

[5] J. G. Proakis and D. K. Manolakis, *Digital Signal Processing: Principles, Algorithms and Applications*. Prentice Hall, 1995.

[6] N. Da Dalt, M. Harteneck, C. Sandner, and A. Wiesbauer, "On the jitter requirements of the sampling clock for analog-to-digital converters," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 49, no. 9, pp. 1354 – 1360, Sep. 2002.

[7] A. H. Sayed, *Adaptive Filters*. John Wiley & Sons, NJ, 2008.

[8] Q. Zou, A. Tarighat, and A. H. Sayed, "Compensation of phase noise in OFDM wireless systems," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5407 –5424, nov. 2007.