This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCAD.2020.2966444, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems

Flicker Noise Formulations in Compact Models

Geoffrey J. Coram, *Senior Member, IEEE*, Colin C. McAndrew, *Fellow, IEEE*, Kiran K. Gullapalli, and Kenneth S. Kundert, *Fellow, IEEE*

***Abstract*—This paper shows how to properly define modulated flicker noise in Verilog-A compact models, and how a simulator should handle flicker noise for periodic and transient analyses. By considering flicker noise in a simple linear resistor driven by a sinusoidal source, we demonstrate that the absolute value formulation used in most existing Verilog-A flicker noise models is incorrect when the bias applied to the resistor changes sign. Our new method for definition overcomes this problem, in the resistor as well as more sophisticated devices. The generalization of our approach should be adopted for flicker noise, replacing the formulation in existing Verilog-A device models, and it should be used in all new models. Since Verilog-A is the de facto standard language for compact modeling, it is critical that model developers use the correct formulation.**

***Index Terms*—Flicker noise, compact models, Verilog-A, modulated stationary noise model.**

I. Introduction

Since the introduction of compact modeling extensions to the Verilog-AMS Language Reference Manual (LRM) [1] version 2.2, in 2004, most compact models for circuit simulation have been developed in Verilog-AMS. More specifically, model developers use the analog-only subset, called Verilog-A. Several papers [2], [3] have offered suggestions on effective coding practice specifically for compact models. The Compact Model Coalition, part of the Silicon Integration Initiative, has been standardizing compact models starting with BSIM3 in 1996; it requires new candidate standard models to be provided in Verilog-A.

The Verilog-A language provides four functions for specifying small-signal noise sources, of which two are of interest for compact models: `white_noise()` and `flicker_noise()`. The LRM describes their behavior for small-signal noise analysis, as implemented in Berkeley SPICE [4] and its descendants, both open-source and commercial.

Beyond the standard small-signal noise analysis, many modern simulators also offer noise analysis for circuits under large-signal time-varying conditions. These analyses include Pnoise (periodic noise) and HBnoise (harmonic balance noise) for periodic behavior, from periodic steady-state (PSS) and harmonic balance (HB) simulations, respectively, and TRnoise (transient noise) for general transient simulations. The analysis algorithms are based on papers from some 25 years ago [5]–[8]. Instead of linearizing around a dc operating point, as in standard noise analysis, Pnoise and HBnoise linearize around a time-varying operating point; TRnoise adds random samples to the large-signal circuit description. All three of these approaches allow accounting for frequency translation of noise, an effect that is not obtained from standard ac noise analysis [7].

G. J. Coram is with Analog Devices, Inc., Wilmington, MA 01887 USA
C. C. McAndrew is with NXP Semiconductors, Chandler, AZ 85224 USA
K. K. Gullapalli is with NXP Semiconductors, Austin, TX 78735 USA
K. S. Kundert is with Designer's Guide Consulting, San Jose, CA, USA

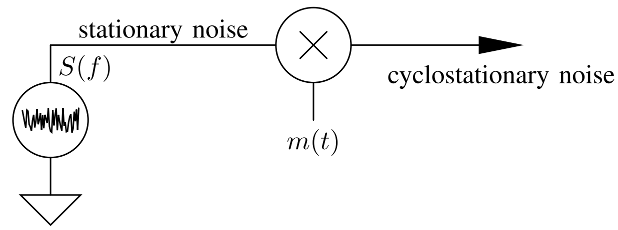

Fig. 1. The creation of cyclostationary noise by a periodically-varying bias $m(t)$ in a modulated stationary noise model.

Most compact device models include noise equations, but the focus is on small-signal noise at dc operating points. Verilog-A implementation of correlated noise has been discussed [9] but, to our knowledge, there is no previous work on flicker noise modeling in Verilog-A for large-signal time-varying simulations. In fact, the Verilog-A noise functions were proposed without consideration of these types of simulations. Here, we show that the standard approach for flicker noise modeling in Verilog-A does not work for large-signal circuit responses, and we present the correct way to model flicker noise for Pnoise, HBnoise, and TRnoise analyses.

II. Modulated Stationary Noise Models

In a traditional SPICE noise analysis the output noise contributed by a component is determined by computing the noise generated by the component and the transfer function from the component to the output. The contribution is the product of the component noise and the transfer function. The total output noise power is the sum of all the contributions from each of the components in the circuit. The noise produced by the component is a function of its operating point (including temperature), which is computed by a dc analysis that precedes the noise analysis.

With a time-varying noise analysis, the noise is computed while the underlying bias point and circuit behavior changes with time. These changes act to modulate the noise contributed by a component. How the noise generated by a component is affected by dynamic changes in its operating point can be difficult to understand and model [10], [11]. However, in many common situations it is possible to assume that the underlying noise process is bias-independent even while allowing the time-varying bias to modulate the noise before it reaches the terminals of the component, as shown in Figure 1.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCAD.2020.2966444, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems

This assumption results in *modulated stationary noise models* [11] that fit very naturally into simulators.[1] For example, consider an RF noise analysis like Pnoise. It works very much like the traditional noise analysis, except the time-varying operating point modulates both the noise produced by the component and the transfer function from the component to the output. With modulated stationary noise models, the noise from the component is easily partitioned into two pieces: the underlying bias-independent noise, represented by its power spectral density $S(f)$, and the modulation of the noise by the component itself, represented by $m(t)$. It is natural and easy for the simulator to combine the modulation from the component, $m(t)$, with the modulated transfer function of the circuit, making the overall computation straightforward.

Modulated stationary noise models work well for modeling the flicker noise in linear resistors, which is due to a fluctuation in the value of the resistance over time [12], [13]. The resistor is linear, so this effect is completely independent of bias. We often think of the flicker noise of a resistor as being produced by a series voltage source or a shunt current source, but this is an approximation that allows us to think of the resistor as time-invariant. Consider a resistor that has resistance of $R + \delta r(t)$ where R is the nominal resistance and δr is the fluctuation of the resistance due to flicker noise. Assume the resistor is driven with a dc current I. Then,

$$V + \delta v(t) = (R + \delta r(t)) \cdot I\,. \tag{1}$$

The resulting noise voltage is

$$\delta v(t) = \delta r(t) \cdot I\,, \tag{2}$$

which includes a bias-independent flicker noise component, δr, and a bias-dependent modulation term, I.

Modulated stationary noise models are not completely general. There are observed circuit behaviors that cannot be modeled using modulated stationary noise sources. For example, the low frequency noise produced by some circuits can be reduced by regularly switching off the bias of the circuit [10]. This effect cannot be accurately modeled using modulated stationary noise sources. Despite these limitations, modulated stationary noise models are suitable in many common situations; and when not completely accurate, often provide a good starting point that is accurate to first order.

Consider flicker noise in MOSFETs. A reasonable first order model is to assume that that the flicker noise is a bias independent variation in the value of the threshold voltage, V_T. Again, because we like time-invariant models, we model the flicker noise by adding a time-varying current source in parallel with the channel, where the current in this noise source is roughly equal to the fluctuation in V_T multiplied by the transconductance of the FET, g_m. This is a modulated stationary noise model where $m = g_\mathrm{m}$ and it is the basic approach that has been used by simulators for many years. While it is not completely accurate for switched-bias circuits, it has proven useful and predictive for a wide variety of other circuits.

[1] In this paper we use the term stationary to denote a noise process where the statistics do not change with time, meaning that the average value (the mean), the power level (the variance) and the correlations (the autocorrelation function) are constant. We use the term cyclostationary to refer to a noise process where these quantities do vary with time, but in a cyclic fashion. We use the terms modulated and non-stationary to refer to noise processes where the statistics do vary with time, so cyclostationary processes are also non-stationary. In addition, a flicker noise process with a true $1/f$ power spectral density is non-stationary because the statistical metrics break down as the period of observation becomes infinite (the variance, or noise power, increases without bound as f goes to zero). However, real devices cannot behave that way. The noise flattens out at low frequencies, making their total noise power finite [12]. We confine ourselves to real devices and so consider a flicker noise process to be stationary or not based only on whether it is modulated.

III. SPICE Flicker Noise Model

If we assume that a resistor that exhibits flicker noise is driven with a constant voltage V, then

$$I + \delta i(t) = \frac{V}{(R + \delta r(t))}\,. \tag{3}$$

By recognizing that $V = IR$ and rearranging the terms this can be written as

$$\delta i(t) = -I \cdot \delta r(t)/R\,. \tag{4}$$

Thus, the power spectral density of the expected noise current is

$$S_{ii} = I^2\, S_{rr} = I^2\, \frac{K}{f}\,. \tag{5}$$

The traditional model used to represent flicker noise of a current I in SPICE is

$$S_{ii} = \frac{\mathtt{KF} \cdot I^{\mathtt{AF}}}{f^{\mathtt{EF}}}\,. \tag{6}$$

SPICE does not model flicker noise for a resistor, but this form is used for the diode, JFET, and MOSFET [4]. This equation matches (5) if $\mathtt{KF} = K$, $\mathtt{AF} = 2$, and $\mathtt{EF} = 1$.

`AF` and `EF` are fitting parameters. There is little to no physical justification for them to be different from their default values of 2 and 1. Nonetheless, they are well established and are routinely set to values different from their defaults.

IV. Modeling Flicker Noise in Verilog-A

A key benefit of Verilog-A is that many analog simulators support dynamic compilation of user-supplied models written in Verilog-A. We are thus able to test out flicker noise models in various simulators without needing to change either the simulators or their built-in models. A simple linear resistor with flicker noise is sufficient to produce our key result.

In Verilog-A, a resistor of value R between nodes `a` and `b` is implemented via

```
Ir      = V(a,b)/R;
I(a,b) <+ Ir;
```

The *contribution operator* `<+` adds a current through the branch connecting the nodes `a` and `b` of value `Ir`. A complete Verilog-A implementation of a resistor can be found in the appendix; sample netlists can be downloaded from [14].

Noise in the resistor can be contributed as well:

```
I(a,b) <+ white_noise(4.0*`P_K*
              $temperature/R, "thermal");
```

where the value of Boltzmann's constant `` `P_K `` is defined in a header file provided in the LRM, and `$temperature` is the circuit analysis temperature in Kelvin. This contribution creates a small-signal noise current source on the branch `(a,b)` with a frequency independent power spectral density whose value is specified by the first argument; the second argument is a name for the simulator to give when presenting results.

Flicker noise for a resistor is usually modeled as being a function of the dc current `Ir` through the resistor

```
Pn       = KF*pow(abs(Ir), AF);
I(a,b) <+ flicker_noise(Pn,            (7)
                  EF, "flicker");
```

where `KF`, `AF`, and `EF` are parameters of the model that are adjusted to fit measured data and `Pn` is the amplitude of the noise power for $f = 1$. `KF` is a prefactor; `AF` is the exponent of the current; and `EF` is the exponent of frequency, which is exactly 1.0 for true $1/f$ noise.

The deterministic current `Ir` (dc or transient) may be positive or negative, and while `AF` is 2 by default, it may be fractional. Trying to compute `pow(Ir, AF)` results in a math error if `Ir` < 0 and `AF` is not an integer. Further, a noise power spectral density must be non-negative. Hence, most models include the `abs()` call in `pow(abs(Ir),AF)` without further thought. Indeed, the Verilog-AMS LRM itself (Section 4.6.4.5 in version 2.4) provides the following expression for flicker noise in a diode:

```
flicker_noise(kf*pow(abs(I(<a>)), af), ef)
```

However, this expression is incorrect and gives rise to peculiar results, as the next section shows.

V. Simulation Results

In [12], analysis of a linear resistor driven by a sinusoidal voltage source of frequency f_0 shows that $1/f$ noise should be a spectrum that varies as $1/|f - f_0|$. Fig. 2 shows Pnoise simulation results of the model (7) compared to the expected reference behavior of [12]. HBnoise simulation results are, to within numerical tolerances, the same as the Pnoise results. Figs. 3 and 4 show TRnoise simulation results for the reference model and the model (7), respectively; they are consistent with the Pnoise and HBnoise results.

Clearly, the model (7) does not work properly for noise under large-signal time-varying conditions when `Ir` changes sign. It incorrectly exhibits a dc-like $1/f$ component, although there is no dc current, it lacks the expected $1/|f - f_0|$ sidebands, and it has components at integer multiples of $2f_0$, which should not be there.

VI. Cosine Mixer Power Spectral Density

In this section, we derive the power spectral density for an ideal cosine mixer. We then extend this in section VII to analyze the incorrect results from the model (7).

The autocorrelation of a random signal $x(t)$ is [15]

$$R_{xx}(t, t + \tau) = E\left[x(t) \cdot x(t + \tau)\right] , \tag{8}$$

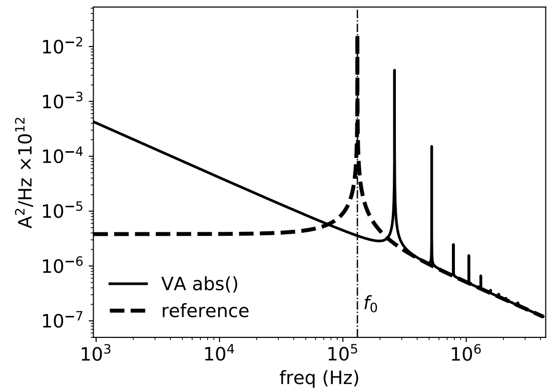

Fig. 2. Pnoise simulation results for a 100 Ω resistor driven by a 0.1 V amplitude sinusoid at $f_0 = 2^{17}$ Hz. `KF`$=10^{-6}$, `AF`$=2$, `EF`$=1$.

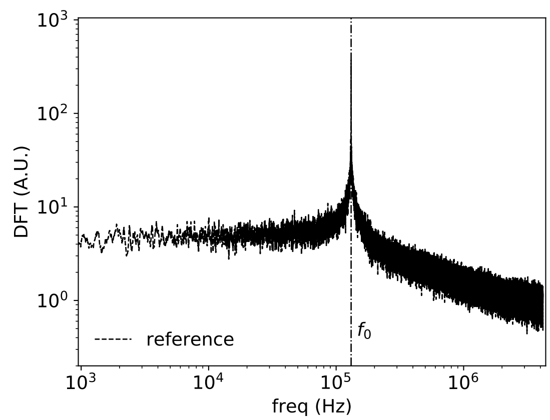

Fig. 3. TRnoise reference simulation results, same device and conditions as Fig. 2.

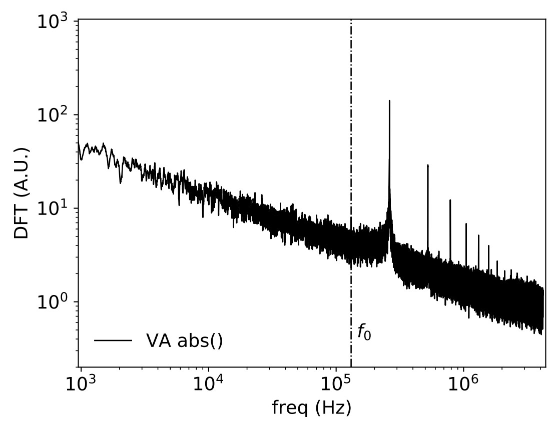

Fig. 4. TRnoise simulation results using the model (7), same device and conditions as Fig. 2.

where $E()$ denotes expectation. $x(t)$ is cyclostationary with period $T = 1/(2\pi f_0)$ if the autocorrelation is invariant to a translation by T, i.e. if

$$R_{xx}(t, t+\tau) = R_{xx}(t+T, t+\tau+T) \,. \tag{9}$$

In this case, we can expand R_{xx} as a Fourier series

$$R_{xx}(t, t+\tau) = \sum_{k=-\infty}^{\infty} R_{x_k}(\tau) \cdot e^{j2\pi k f_0 t} \,, \tag{10}$$

where the Fourier coefficients, which [7] calls "harmonic autocorrelation functions," are

$$R_{x_k}(\tau) = \frac{1}{T} \int_0^T R_{xx}(t, t+\tau) \cdot e^{-j2\pi k f_0 t} \, dt \,. \tag{11}$$

The "harmonic power spectral densities" [7] are the Fourier transforms of these harmonic autocorrelation functions:

$$S_{x_k}(f) = \int_{-\infty}^{\infty} R_{x_k}(\tau) \cdot e^{-j2\pi f \tau} \, d\tau \,. \tag{12}$$

Consider a sinusoidal mixer having unity amplitude and frequency f_0, with input $x(t)$ and output $y(t)$,

$$y(t) = \cos(2\pi f_0 t) \cdot x(t) \,. \tag{13}$$

If the signal $x(t)$ contains noise we want to determine the power spectral density of the output signal $y(t)$, which is cyclostationary as long as $x(t)$ is stationary or cyclostationary with period $1/f_0$. Further, from (13), (8), and (10)

$$\begin{aligned}
&R_{yy}(t, t+\tau) \\
&= E\left\{\cos(2\pi f_0 t) \cdot x(t) \cdot \cos[2\pi f_0 (t+\tau)] \cdot x(t+\tau)\right\} \\
&= \cos(2\pi f_0 t) \cdot \cos[2\pi f_0 (t+\tau)] \cdot E\left[x(t) \cdot x(t+\tau)\right] \\
&= \cos(2\pi f_0 t) \cdot \cos[2\pi f_0 (t+\tau)] \cdot R_{xx}(t, t+\tau) \\
&= \cos(2\pi f_0 t) \cdot \cos[2\pi f_0 (t+\tau)] \cdot \sum_{k=-\infty}^{\infty} R_{x_k}(\tau) e^{j2\pi k f_0 t} \,,
\end{aligned} \tag{14}$$

so the harmonic autocorrelation functions of y are

$$\begin{aligned}
R_{y_i}(\tau) = &\frac{1}{T} \int_0^T \Bigg\{ \cos(2\pi f_0 t) \cdot \cos[2\pi f_0 (t+\tau)] \\
&\times \sum_{k=-\infty}^{\infty} R_{x_k}(\tau) e^{j2\pi (k-i) f_0 t} \Bigg\} \, dt \,.
\end{aligned} \tag{15}$$

The harmonic power spectral densities of y are then

$$\begin{aligned}
&S_{y_i}(f) \\
&= \int_{-\infty}^{\infty} \Bigg\{ \frac{1}{T} \int_0^T \cos(2\pi f_0 t) \cdot \cos[2\pi f_0 (t+\tau)] \\
&\qquad \times \sum_{k=-\infty}^{\infty} R_{x_k}(\tau) e^{j2\pi (k-i) f_0 t} \, dt \Bigg\} \cdot e^{-j2\pi f \tau} \, d\tau \\
&= \sum_{k=-\infty}^{\infty} \frac{1}{T} \int_0^T \cos(2\pi f_0 t) \cdot e^{j2\pi (k-i) f_0 t} \\
&\times \left\{ \int_{-\infty}^{\infty} \cos[2\pi f_0 (t+\tau)] \cdot e^{-j2\pi f \tau} \cdot R_{x_k}(\tau) \, d\tau \right\} \, dt \,,
\end{aligned} \tag{16}$$

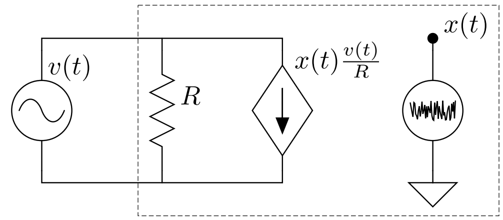

Fig. 5. Resistor model that includes modulated flicker noise.

where the integration and summation order was changed.

Now $\cos(a) = 0.5 \cdot (e^{ja} + e^{-ja})$, so the inner integral over $d\tau$ in (16) is

$$\begin{aligned}
&0.5 \int_{-\infty}^{\infty} \left[e^{j2\pi f_0 (t+\tau)} + e^{-j2\pi f_0 (t+\tau)} \right] \cdot e^{-j2\pi f \tau} \cdot R_{x_k}(\tau) \, d\tau \\
&= 0.5 \cdot e^{j2\pi f_0 t} \int_{-\infty}^{\infty} e^{-j2\pi (f - f_0)\tau} \cdot R_{x_k}(\tau) \, d\tau \\
&\quad + \, 0.5 \cdot e^{-j2\pi f_0 t} \int_{-\infty}^{\infty} e^{-j2\pi (f + f_0)\tau} \cdot R_{x_k}(\tau) \, d\tau \\
&= 0.5 \cdot e^{j2\pi f_0 t} S_{x_k}(f - f_0) + 0.5 \cdot e^{-j2\pi f_0 t} S_{x_k}(f + f_0) \,.
\end{aligned} \tag{17}$$

Using this in (16) and replacing $\cos(2\pi f_0 t)$ by its complex exponential form gives

$$\begin{aligned}
S_{y_i}(f) = &\sum_{k=-\infty}^{\infty} \frac{1}{4T} \int_0^T \\
&S_{x_k}(f - f_0) \left[e^{j2\pi (k-i+2) f_0 t} + e^{j2\pi (k-i) f_0 t} \right] \\
&+ S_{x_k}(f + f_0) \left[e^{j2\pi (k-i-2) f_0 t} + e^{j2\pi (k-i) f_0 t} \right] \, dt \,.
\end{aligned} \tag{18}$$

Because of the periodicity of the complex exponential,

$$\frac{1}{T} \int_0^T dt \; e^{-j2\pi k f_0 t (m-n)} = \delta_{m,n} \,, \tag{19}$$

where $\delta_{m,n}$ is the Kronecker delta, only four terms of the summation in (18) are non-zero, for $k = i, i \pm 2$, so

$$\begin{aligned}
S_{y_i}(f) = &\frac{1}{4} \left[S_{x_{i-2}}(f - f_0) + S_{x_i}(f - f_0) \right. \\
&\left. + S_{x_i}(f + f_0) + S_{x_{i+2}}(f + f_0) \right] \,,
\end{aligned} \tag{20}$$

which is consistent with (3) of [7]. For a stationary input, S_{x_k} is non-zero only for $k = 0$, and we typically measure the average output noise S_{y_0}, so this becomes

$$S_{y_0}(f) = \frac{S_{x_0}(f - f_0) + S_{x_0}(f + f_0)}{4} \,. \tag{21}$$

This means that the noise spectrum of $x(t)$ is frequency shifted by the mixer to $\pm f_0$. This is precisely the behavior of the reference results in Figs. 2 and 3, and it matches the results in [12], obtained by a different method.

VII. Modulated Flicker Noise

Consider the circuit in Fig. 5, which has an independent voltage source $v(t)$ driving a resistor, and a dependent current

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCAD.2020.2966444, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems

source whose value is the product of current in the resistor and the noise voltage $x(t)$.

This equivalent circuit is a modulated stationary noise model that separates the flicker noise process from the bias dependence in a way that makes the analysis more tractable and bypasses questions about whether the bias dependence affects the statistics of the random process [7]. The random process is bias-independent and stationary, and the bias dependence is introduced by the deterministic modulation. This interpretation aligns with van der Ziel's description of flicker noise as a fluctuation of the resistance and the current as a method of detecting it [12]; when the current is time-varying, it modulates the observed noise spectrum; see (5).

If $v(t)$ is a dc source, i.e. $v(t) = V$, then the dependent source provides a constant gain of V/R to the noise source, and the the noise current power spectral density is the product of $(V/R)^2$ and the power spectral density of $x(t)$. If $x(t)$ is a flicker noise source with power spectral density $S_{xx}(f) = \mathtt{KF}/f$, then the current noise power spectral density is

$$S_{ii}(f) = \left(\frac{V}{R}\right)^2 \frac{\mathtt{KF}}{f}\,. \tag{22}$$

This is just (7) with $\mathtt{AF} = 2$ and $\mathtt{EF} = 1$, and is equivalent to the result (5) derived from a resistance fluctuation viewpoint. For a dc bias, the absolute value does not matter; the autocorrelation (and hence power spectral density) depends on the square of the signal, so the gain is squared, eliminating any minus sign.

If the voltage source is a cosine source, $v(t) = \cos(2\pi f_0 t)$, the situation changes. When the dependent source multiplies the noise by the current, we recover the mixer situation analyzed in the previous section, and we get the familiar output noise power spectral density, consisting of copies of the $1/f$ spectrum shifted to $\pm f_0$.

However, if the dependent source multiplies the noise by the absolute value of the current, so $i(t) \rightarrow x(t) \cdot |v(t)/R|$, the analysis of the previous section breaks down. Eq. (16) becomes

$$S_{y_i}(f) = \sum_{k=-\infty}^{\infty} \frac{1}{T}\int_0^T |\cos(2\pi f_0 t)| \cdot e^{j2\pi(k-i)f_0 t} \times \left\{ \int_{-\infty}^{\infty} |\cos[2\pi f_0(t+\tau)]| \cdot e^{-j2\pi f\tau} \cdot R_{x_k}(\tau)\, d\tau \right\} dt\,. \tag{23}$$

To evaluate (23) we need to introduce the Fourier series of $|\cos(2\pi f_0 t)|$. This is

$$|\cos(2\pi f_0 t)| = \sum_{k=-\infty}^{\infty} a_k \cdot e^{j2\pi f_0 k t} \tag{24}$$

where

$$a_k = \frac{1}{T}\int_0^T |\cos(2\pi f_0 t)| \cdot e^{-j2\pi f_0 k t}\, dt = \begin{cases} \dfrac{2\cdot(-1)^{k/2}}{\pi\cdot(1-k^2)} & \text{for even } k \\ 0 & \text{for odd } k \end{cases}\,. \tag{25}$$

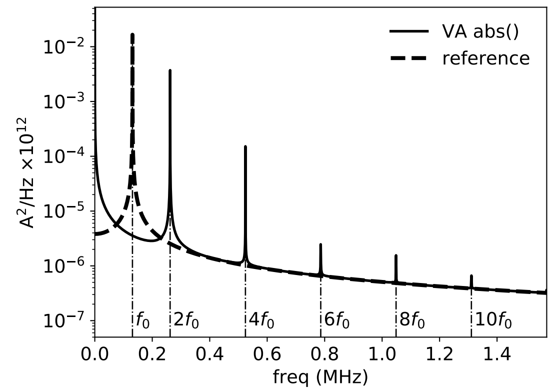

Fig. 6. Results of Fig. 2 with linear abscissa scale.

Intuitively, if we modulate the noise by $|\cos(2\pi f_0 t)|$, we would expect a dc component as well as even harmonics, but no odd harmonics, and (25) verifies that these are the components selected from (23). This is also precisely the behavior of the incorrect results in Figs. 2 and 4 from the Verilog-A noise model of (7) that uses `abs(Ir)`. Fig. 6 show the results of Fig. 2 with a linear abscissa scale, which makes the frequencies of the harmonics more apparent.

We note that (16) and (23) are identical when x is white noise, uncorrelated in time ($R_{xx}(t, t+\tau) = \delta(\tau)$, where here $\delta(\tau)$ is the Dirac delta function, which means that $R_{x_0}(t, t+\tau) = \delta(\tau)$, and the other harmonic autocorrelation functions $R_{x_k}(\tau) = 0$, $k \neq 0$). Accounting for the sign of the modulating signal is necessary only for colored noise.

Consider also the scenario where $v(t)$ is a square wave: If the noise is determined by the absolute value of the modulation, then this situation would be indistinguishable from a dc bias (other than the short transition times).

VIII. Extracting the Modulation Function

How did this happen? How did we end up taking the absolute value of the modulating signal? It is easy to say that the problem is coming from the use of the absolute value function in (7). While that is certainly true, there is more to it. Consider the simpler case of where $\mathtt{AF}$ is fixed to 2 as in (6).

$$S_{ii} = I^2 \frac{K}{f}\,. \tag{26}$$

Decomposing this into a modulated stationary noise model gives $S = K/f$ and $m = \sqrt{(I^2)} = |I|$. The square root is required to convert I^2 from a power. Remember that each positive number has two square roots, one positive and one negative. In this case we chose the positive one, but we could have easily chosen the negative one. At the time we were assuming dc conditions, so the sign is irrelevant because m is multiplied by values from a stochastic process with zero mean. However, when the operating point changes with time we have to be more careful. We can see from (4) that the modulation function is $m = -I$, but as before the minus sign is irrelevant, so we can use either $\pm I$. However, if we always choose the positive root for m, even when I changes sign, this is in effect alternating which root is chosen as a function of time, which is not permissible. The fundamental problem is that the amplitude of the flicker noise is specified as a power, which means the sign of the modulation function m is lost.

One way to solve this problem is simply make the noise power argument of the flicker noise function constant and instead modulate the output of the function. This is natural if you know the modulation function. For example, for this simple case we know $m = -I$ and so we can model the noise with:

```
I(a,b) <+ -Ir*flicker_noise(K,
                   EF, "flicker");
```
(27)

However, in many cases it is easier to work with the noise power because that is what is readily available, as with (6). That case is considered next.

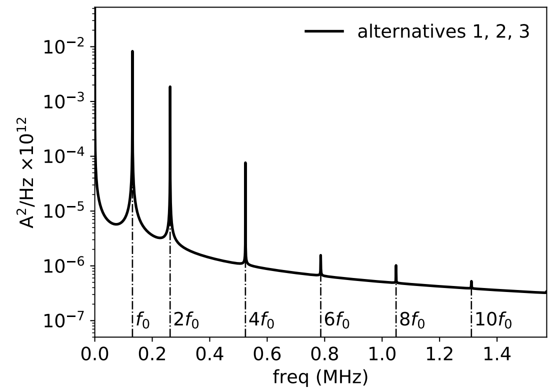

Fig. 7. Pnoise simulation results, alternative noise model implementations 1, 2, and 3, same device and conditions as Fig. 2.

IX. Revised Verilog-A Implementation

As described in the Section VII, periodic and transient noise analyses use the modulated stationary noise model [7]. The noise input $u(t)$ is expressed as $u(t) = m(t) \cdot u_s(t)$ where $m(t)$ is the modulation and u_s is stationary noise. The PSD of u, S_{uu}, then involves M_i the Fourier coefficients of $m(t)$. Without going into the details that be found in [7], we note here that when $S_{u_s u_s}$ is frequency independent (white), the sign of $m(t)$ ends up not being necessary. However, for flicker and colored noise sources, we must preserve information about the sign of $m(t)$.

We would like to convey the sign of $m(t)$ to the various noise analyses, by choosing an alternative expression when `Ir` is negative (`Pn` is defined in (7))

```
if (Ir >= 0) begin
  I(a,b) <+  flicker_noise( Pn,
                   EF, "flicker" );
end else begin
  // alternative expression goes here
end
```

Several alternatives appear possible for a Verilog-A model to convey the sign of $m(t)$ to the various noise analyses:

```
// Alternative 1
// add minus sign in first argument
I(a,b) <+  flicker_noise(-Pn,
                   EF, "flicker" );

// Alternative 2
// add leading minus sign
I(a,b) <+ -flicker_noise( Pn,
                   EF, "flicker" );

// Alternative 3
// swap terminal order
I(b,a) <+  flicker_noise( Pn,
                   EF, "flicker" );
```

For any dc bias condition, we expect all of these implementations to give the same small-signal noise as (7), because only one of the if clauses is active. During a transient analysis or periodic excitation where `Ir` changes sign, we expect to obtain different noise results. Unfortunately, the results are not what we want. These three alternatives each describe *two* noise sources, one of which is active for positive `Ir` (and zero for $\texttt{Ir} < 0$), and the other is active for negative `Ir` (and zero for $\texttt{Ir} \geq 0$). Since their names are the same, the contributions are combined in the noise summary report, per section 4.6.4 of the Verilog-AMS LRM [1], but the noise sources are independent, and hence uncorrelated. Each noise source is modulated by a half-wave rectified cosine. The Fourier coefficients for the component that is nonzero in the first and fourth quarter cycles is

$$a_k = \begin{cases} \dfrac{(-1)^{k/2}}{\pi \cdot (1-k^2)} & \text{for even } k \\ 1/4 & \text{for } k = 1 \\ 0 & \text{for odd } k > 1 \end{cases} \tag{28}$$

and a similar set of coefficients apply to the component that is nonzero in the second and third quarter cycles. Besides the dc and even harmonic components, as for the absolute value modulation case, (28) indicates there should also be a component at the fundamental frequency f_0. Fig. 7 shows simulation results from these three alternatives, which are identical; the additional component at f_0 is clear compared to Fig. 6, but they do not give the correct results (the "reference" curve of Fig. 2).

We must instead use one of the following two alternatives, which have only one `flicker_noise` call and hence one flicker noise source.

```
// Alternative 4
// change sign of argument
if (Ir < 0) begin
  Pn = -Pn;
end
I(a,b) <+  flicker_noise( Pn,
                   EF, "flicker" );

// Alternative 5
// add prefactor of +/- 1
integer sgn;
```

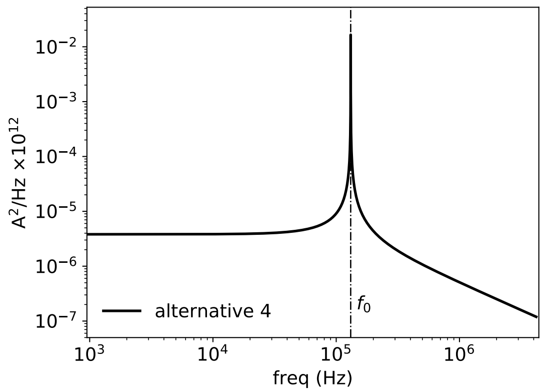

Fig. 8. Pnoise simulation results, alternative noise model implementation 4 or 5, same device and conditions as Fig. 2.

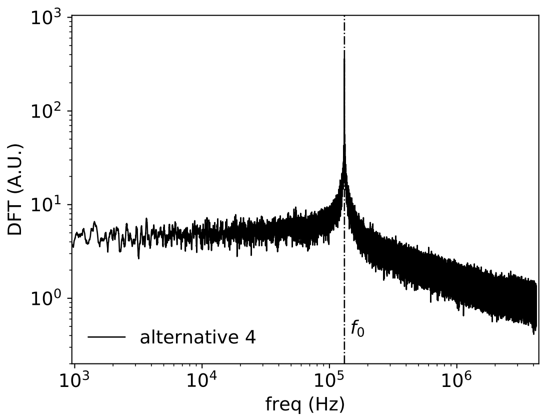

Fig. 9. TRnoise simulation results, alternative noise model implementation 4 or 5, same device and conditions as Fig. 2.

```
sgn = (Ir >= 0) ? +1 : -1;
I(a,b) <+ sgn * flicker_noise( Pn,
                EF, "flicker" );
```

It may seem peculiar that the noise power argument `Pn` in alternative formulation 4 is negative for $\texttt{Ir} < 0$, but the LRM does not require the first argument of `flicker_noise` to be non-negative. Allowing negative values for `Pn` in the integrals of (16), instead of using their absolute values, changes the results of the integration. In effect, when implemented properly, the simulator is allowing you to choose the root it uses for $m(t)$ when computing $m = \sqrt{\texttt{Pn}}$. If `Pn` is positive, the positive root is chosen, if `Pn` is negative, then the negative root is chosen.

Alternative 5 is actually a modification of the approach found in (27), where only the sign (± 1) of the modulation is used as a prefactor.

Fig. 8 shows Pnoise simulation results for these two alternatives, which are identical. Clearly, this behavior matches the $\pm f_0$ frequency shifted noise predicted by (21) and the reference results of Fig. 2. TRnoise simulation results based these alternative implementations also match those of the reference model; compare Figs. 9 and 3. The appendix contains complete Verilog-A code for a resistor with flicker noise using Alternative 4.

Note that some simulators may not properly simulate one or the other of these alternatives, as will be discussed in Section XI. There is another alternative formulation, which requires an internal node:

```
// Alternative 6
// use internal node
electrical noi;

// stationary flicker noise
I(noi) <+ flicker_noise(KF, EF,
                     "flicker");
// convert noise current to voltage
I(noi) <+ V(noi);

// modulate noise
I(a,b) <+ Ir * V(noi);
```

This alternative closely follows (27) and has been tested in several commercial simulators. However, it requires an extra internal node for each flicker noise source.

X. Other Device Models

The noise power formulation for flicker noise is common, examples from bipolar and diode models include:

- Mextram: (version 504.12)

```
powerFBC1fB1 = KF_M * (1.0-XIBI)
     * pow((abs(Ib1)/(1.0-XIBI)), AF);
I(b2,e1) <+ flicker_noise(
     powerFBC1fB1, 1);
```

- HiCuM: (version 2.4.0)

```
flicker_Pwr = kf
     * pow(abs(ibei+ibep),af);
I(br_biei) <+ flicker_noise(
     flicker_Pwr,1.0, "flicker");
```

- Diode_CMC:

```
jfnoise = KF_i
     * pow(abs(ijun)*MULT_i, AF_i);
I(A, AIK) <+ flicker_noise(
     jfnoise, 1.0, "flicker");
```

Bipolar and diode currents, and therefore flicker noise, are generally negligible under reverse bias. The contribution from the forward-biased portion of a cycle dominates the output noise, so using the `abs()` formulation for flicker noise is of little consequence in practice.

However, field-effect transistors, particularly those biased around $V_{ds} = 0$ such as switches or passive mixers, are susceptible to the issue. As an example of the confusion created by improperly handling flicker noise in cyclostationary noise analysis, Redman-White and Leenaerts use erroneous simulation results (shown in their Figure 2) to support their observations of low frequency noise in passive mixers [16]. In general, FET models detect if V_{ds} is negative, and if so the drain and source terminals are interchanged, calculations are

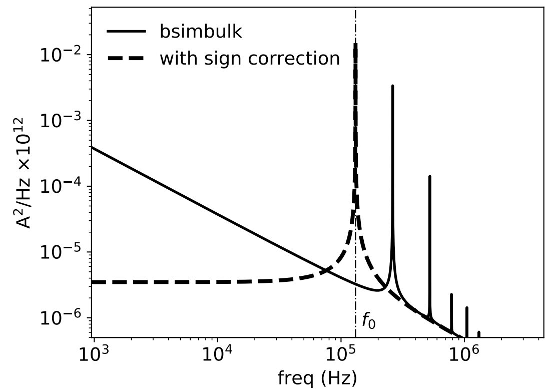

Fig. 10. Pnoise simulation results for a BSIMBULK transistor.

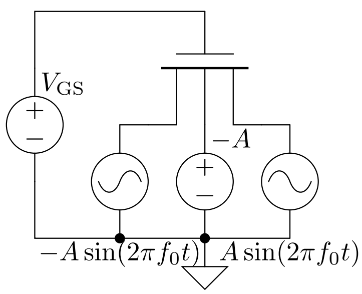

Fig. 11. Schematic for BSIMBULK and PSP test.

done as if $V_{ds} \geq 0$, which gives non-negative values for both I_{ds} and flicker noise. The sign of I_{ds} is then reversed, but this is *not* done for the flicker noise, so the result is equivalent to the `abs()` formulation of (7).

Fortunately, most FET models include a variable, named `sigvds` or `sigVds`, that is 1 if $V_{ds} \geq 0$ and -1 otherwise. This enables a simple fix for the problem. For example, for BSIMBULK [17]

```
I(di,si) <+ flicker_noise(
    FNPowerAt1Hz, EF, "flicker");
```

To include the sign flip for reverse mode this just needs to be changed to

```
I(di,si) <+ flicker_noise(
    sigvds*FNPowerAt1Hz, EF, "flicker");
```

Fig. 10 shows the effectiveness of this simple change (the flicker noise parameters for BSIMBULK were adjusted to closely match Fig. 2, and the source and drain were driven differentially, to avoid harmonic generation from non-symmetric drain current between $V_{ds} > 0$ and $V_{ds} < 0$ portions of the cycle). Fig. 11 shows the schematic for the simulation.

For the PSP 103.6.0 model [18],

```
I(DI,SI) <+ flicker_noise(
    MULT_i*Sfl, EF_i, "flicker");
```

and simply changing this to

```
I(DI,SI) <+ flicker_noise(
    sigVds*MULT_i*Sfl, EF_i, "flicker");
```

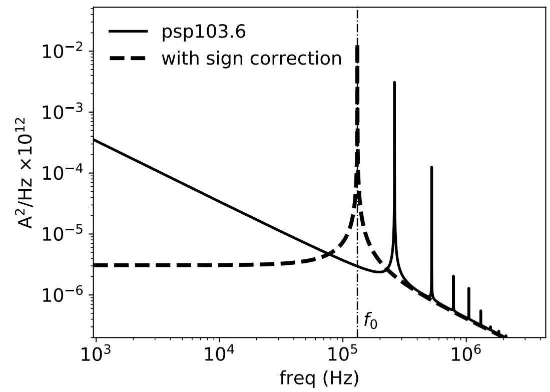

Fig. 12. Pnoise simulation results for a PSP transistor.

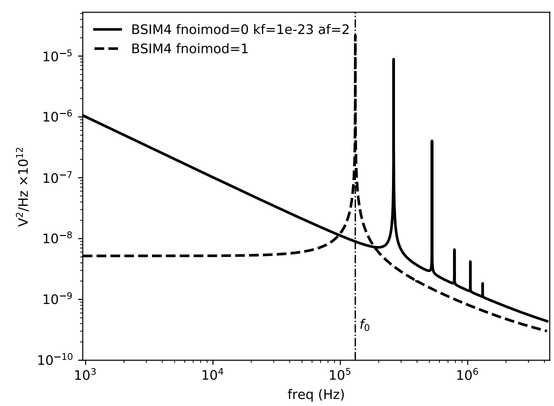

Fig. 13. Pnoise simulation results for a BSIM4 transistor.

completely solves the issue of large-signal time-varying noise simulation for PSP. Fig. 12 shows the effect of this change.

The error in flicker noise formulations is not confined to models written in Verilog-A. In tests with a commercial simulator, we found an error in BSIM4, which is distributed as C code [19]. When using FNOIMOD=0, an incorrect spectrum is obtained when simulating the schematic of Fig. 11, and yet the correct spectrum is obtained for FNOIMOD=1, as shown in Fig. 13.

XI. Simulator Limitations

We have recommended two basic approaches to modeling flicker noise in Verilog-A that are suitable for simulations involving time-varying operating points. In the first, we assume that the modulation function is readily available. In this case you would pass a constant value to the noise power argument of the flicker noise function and multiply its output by the modulation function. An example that is suitable for (4) and given previously in (27) is

```
I(a,b) <+ -Ir*flicker_noise(K,
                 EF, "flicker");
```
(29)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCAD.2020.2966444, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems

In the second, we assume that the instantaneous noise power is readily available. In this case you need to carefully specify the sign of the instantaneous noise power to convey the sign of $m(t)$. An example that is suitable for (6) is given as Alternative 4:

```
if (Ir < 0) begin
  Pn = -Pn;
end
I(a,b) <+  flicker_noise( Pn,
                  EF, "flicker" );
```

Alternative 5 is essentially a hybrid of these two approaches, suitable when the modulation function is not available, but its sign can be determined, and you dislike changing the sign of the noise power argument.

Unfortunately, some simulators do not allow the small-signal stimulus functions such as `flicker_noise()` to be used in expressions, which prevents use of the first approach (as well as the hybrid approach). Some simulators do not allow or do not properly process the sign to the noise power argument, which prevents use of the second approach. And some simulators do not support either, making them unsuitable for simulating flicker noise with time-varying operating points.

Of these limitations, the first is the most problematic. It affects the ability to modulate any small-signal stimulus function, such as `ac_stim` or the `noise_table` functions. For example, the oscillator model in Table 1 of [6] modulates the output of the `flicker_noise` function in order to model the phase noise of the oscillator. That model, and models like it, cannot be used in simulators with this restriction. This restriction also interferes with the ability to generate correlated noise (see Section 4.6.4.6 of [1]). Finally, it runs counter to the language definition of Verilog-A [1] and so causes compatibility issues between simulators.

XII. Conclusion

We have shown a modulated stationary noise model can accurately represent flicker noise in circuits with time-varying operating points. In this case the flicker noise is partitioned into two pieces, a stationary or bias-independent flicker noise process and a modulation function that modulates the output of the flicker noise process to produce a cyclostationary or non-stationary result. However, we also showed that if the time-varying amplitude of the model is specified as an instantaneous noise power, care must be taken when extracting the modulation function to assure that the correct sign is maintained throughout the simulation. Two different approaches were presented to do so. Either will work and which approach is used should be decided based on which results in the simplest representation in Verilog-A.

Model developers need to be aware of the modulated stationary noise model assumption and the modulation function itself. They should also be aware of the two different styles of modeling modulated flicker noise (direct modulation and instantaneous noise power), and they should choose the simplest approach recognizing that if they choose to use instantaneous noise power, they need to pass in the sign of the modulation function as the sign of the instantaneous noise power. Most existing models are formulated using instantaneous noise power but neglect to correct the sign of the power and so need to be updated.

Finally, simulator vendors need to enhance their implementations of Verilog-A to support both approaches.

Appendix: Complete Resistor Model

Here we present complete Verilog-A code for the resistor, for the reader interested in running some experiments. As noted above, not all simulators allow a negative power as the first argument to `flicker_noise`. The original flicker noise formulation may be tested by commenting out the `` `define `` in the first line. Commenting out the thermal noise contribution may also help isolate the flicker noise effects.

```
`define FIXED_MODEL
`include "disciplines.vams"
`include "constants.vams"

module res_va(a,b);
  inout a, b;
  electrical a, b;
  parameter real R  = 100 from (0.0:inf);
  parameter real KF = 1u  from [0.0:inf);
  parameter real AF = 2.0 from (0.1:inf);
  parameter real EF = 1.0 from (-inf:inf);

  analog function integer sign;
    input arg;
    real arg;
`ifdef FIXED_MODEL
    sign = arg >= 0 ? +1 : -1;
`else
    sign = +1;
`endif
  endfunction

  analog begin : vaResistor
    real Ir, Pn;
    Ir = V(a,b)/R;
    I(a,b) <+ Ir;
    I(a,b) <+ white_noise(4.0*`P_K*
                  $temperature/R, "thermal");
    Pn = KF*pow(abs(Ir), AF);
    I(a,b) <+ flicker_noise(sign(Ir)*Pn,
                     EF, "flicker");
  end
endmodule
```

Acknowledgments

We would like to acknowledge Roberto Maurino of Analog Devices, who first observed that the R3_CMC model was upconverting flicker noise to double the modulation frequency, which led to the investigation reported here.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCAD.2020.2966444, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems

REFERENCES

[1] *Verilog-AMS Language Reference Manual*. [Online]. Available for download from Accellera Organization, Inc. at https://accellera.org/downloads/standards/v-ams

[2] G. J. Coram, "How to (and how not to) write a compact model in Verilog-A," in *Proc. IEEE Behav. Model. Simulat. Workshop*, San Jose, CA, USA, Oct. 2004, pp. 97-106.

[3] C. C. McAndrew, et al., "Best practices for compact modeling in Verilog-A," *IEEE J. Electron Dev. Soc.*, vol. 3, no. 5, pp. 383-396, 2015.

[4] L. W. Nagel, "SPICE2: A computer program to simulate semiconductor circuits," Dept. Electr. Eng. Comput. Sci., Univ. California, Berkeley, CA, USA, Tech. Rep. ERL-M520, May 1975.

[5] M. Okumura, H. Tanimoto, T. Itakura, and T. Sugawara, "Numerical noise analysis for nonlinear circuits with a periodic large signal excitation including cyclostationary noise sources," *IEEE Trans. Circuits Syst.-I*, vol. 40, pp. 581-590, Sept. 1993.

[6] K. Kundert, "Introduction to RF simulation and its application," *IEEE J. Solid-State Circuits*, vol. 34, no. 9, pp. 1298-1319, September 1999.

[7] J. Roychowdhury, D. Long, P. Feldmann "Cyclostationary noise analysis of large RF circuits with multitone excitations," *IEEE J. Solid-State Circuits*, vol. 33, pp. 324-336, March 1998.

[8] P. Bolcato and R. Poujois, "A new approach for noise simulation in transient analysis," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 887-890, May 1992.

[9] C. C. McAndrew, G. Coram, A. Blaum, and O. Pilloud, "Correlated noise modeling and simulation," *Proc. Tech. Workshop Compact Modeling*, pp. 40-45, May 2005.

[10] A. van der Wel, E Klumperink, J. Kolhatkar, E. Hoekstra, M. F. Snoeij, C. Salm, H. Wallinga, and B. Nauta. "Low-frequency noise phenomena in switched MOSFETs," *IEEE Journal of Solid-State Circuits*, vol. 4. no. 3, March 2007.

[11] A. G. Mahmutoglu and A. Demir, "Non-Monte Carlo analysis of low-frequency noise: exposition of intricate nonstationary behavior and comparison with legacy models," *Proc. IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 35, no. 11. pp. 1825-1835, Nov. 2016.

[12] A. van der Ziel, *Noise in Solid State Devices and Circuits*, New York: John Wiley & Sons, 1986.

[13] R. Brederlow, W. Weber, C. Dahl, D. Schmitt-Landsiedel, and R. Thewes, "Low-frequency noise of integrated poly-silicon resistors," *Proc. IEEE Trans. Electron Devices*, vol. 48, no. 6, pp. 1180-1187, June 2001.

[14] Netlists for the simulations performed for this paper can be downloaded from The Designer's Guide at https://designers-guide.org/verilog-ams/semiconductors/flicker-noise/flicker-noise.tgz (or .zip)

[15] K. S. Shanmugan and A. M. Breipohl, *Random Signals: Detection, Estimation and Data Analysis*, New York: John Wiley & Sons, 1988.

[16] M. Redman-White and D. M. W. Leenaerts, "$1/f$ noise in passive CMOS mixers for low and zero IF integrated receivers," *Proc. Eur. Solid-State Circuits Conf.*, pp. 375-378, Sept. 2001.

[17] BSIM-BULK 106.2.0 [Online]. Available for download from the University of California, Berkeley at http://bsim.berkeley.edu/models/bsimbulk/

[18] PSP 103.3.0 [Online]. Available for download from the Leti technology research institute of the French Alternative Energies and Atomic Energy Commission (LETI-CEA) at http://www.cea.fr/cea-tech/leti/pspsupport/Documents/psp103p6_vacode.zip

[19] BSIM4 version 4.8.1 [Online]. Available for download from the University of California, Berkeley at http://bsim.berkeley.edu/models/bsim4/