Electrical and Electronic Engineering 2020, 10(1): 15-22
DOI: 10.5923/j.eee.20201001.03

Energy Consumption of a Sub-threshold Analog Signed Multiplier Compared to a Digital Implementation

Grant S. Christiansen*, Gary S. Delp, Barry K. Gilbert

Special Purpose Processor Development Group, Mayo Clinic, Rochester, Minn, USA

Abstract We compare the energy consumed by 8-bit x 8-bit and 16-bit x 16-bit multipliers composed of small analog multipliers implemented using transistors operating in their sub-threshold regions to the energy consumed by equivalent digital implementations. The analysis shows that the analog energy consumption is determined by the required signal-to-noise ratio of the individual analog multipliers and that the energy consumption is higher than the equivalent digital implementation's energy consumption.

Keywords Analog computing, Sub-threshold design, Energy efficiency

1. Introduction

Analog computing, which uses electrical circuits with continuous signals to perform mathematical operations, differs from digital computing in that quantities are represented as signal levels continuously varying over time as opposed to numeric (two-level binary) quantized representations that are sampled over time. Information in analog signals can be stored and transferred in many forms, including voltage levels, current flow, frequencies, or phase-shifted waves.

Analog computing, using transistors in their sub-threshold region of operation, has shown energy consumption advantages in inexact applications, where precise answers are not required (e.g., speech recognition [1], solving differential equations [2]). Given those successes in inexact computing, it appears worth considering whether those energy advantages could be applied to precise mathematical computations consisting of exact multiplications and additions.

As an indicator to whether analog computations can be more energy efficient than digital computations, we consider the energy consumption of an analog signed multiplier using transistors operating in their sub-threshold region, and compare it to the energy consumption of a signed digital multiplier (as would be available in commonly available microprocessors), both implemented in state-of-the-art CMOS technology. The layout of the paper is as follows.

In section 2 we analyze the energy consumption of a signed analog multiplier.

In section 3 we anayze the energy consumption of an equivalent digital signed multiplier.

In section 4 we compare the two types of multipliers and present our conclusions.

2. Analog Multiplier Analysis

2.1. Analog Design Challenges

Challenges with analog circuitry include, e.g., signal linearity, sensitivity to noise, transistor mismatch, drift, component tolerances, and offset and gain variation with temperature, process, and supply. Adding compensation circuitry may increase energy consumption.[1]

To attempt to address the transistor mismatch and tolerance concerns, this paper will examine an analog implementation of signed multipliers using floating-gate transistors, described in the next section. We also will use transistors in their sub-threshold region, which has the promise of lower power consumption and improved linearity of the multiplier circuit compared to using transistors in their active region.

2.2. Sub-Threshold Design and Floating Gate Design

Traditionally analog circuitry is designed using transistors biased into their "active" region. One idea to reduce power consumption is to construct circuits biased in their "sub-threshold" region. However, "sub-threshold" designs

* Corresponding author:
christiansen.grant@mayo.edu (Grant S. Christiansen)
Published online at http://journal.sapub.org/eee

1 In this paper we have been careful to distinguish energy from power (energy over time). The static power is amortized over a larger number of operations when the operations/unit time is increased; this can reduce the energy per operation.

suffer from degraded transistor-to-transistor matching and noise performance (since the circuit current magnitudes would be less) compared to "active" designs.

Two techniques have been suggested to improve matching in sub-threshold designs: adaptive body biasing (ABB) [3] and the use of floating gate (FG) transistors [4]. In ABB, additional circuitry is added to measure and compensate for any mismatches. Since the additional circuitry would add to the power consumption, we decided to focus on the second, floating gate, method, which does not have the power penalty of adaptive circuitry.

A "floating gate" transistor is a FET manufactured so its gate terminal is insulated from the rest of the transistor, forming a gate that, while "floating" from the transistor channel, is capacitively coupled to it. Additional capacitors are added to the structure to allow for connection to a circuit (isolating the programmed floating gate from the rest of the circuit) for programming. Programming injects or removes charge on the floating gate, which causes a change in the effective threshold voltage. The result is that any variance in the threshold voltages can be minimized by injecting or removing charge on the floating gate [5].

2.3. High-Level Analog Multiplier Architecture

One challenge of analog computing is the very wide dynamic range needed to represent numbers made up of only a small number of bits. Representing a 16-bit number, for example, as an analog voltage between zero and 5 volts requires a resolution of 76 µV. Fortunately, there are techniques (e.g., breaking the problem down into smaller segments with fewer bits each using residue number systems for the calculations) that allow a higher resolution result to be built up from the results of many lower-resolution calculations; it is therefore possible to use multiple analog circuits of reasonable resolutions.

Figure 1 shows a possible design of an *L*-bit by *L*-bit signed multiplier (producing a result of width 2*L* bits) that addresses the resolution challenge by decomposing it into *M* *N*-bit x *N*-bit signed multipliers implemented as Gilbert-cell multipliers [6].

The logic block at the left side of Figure 1 splits the larger *L*-bit x *L*-bit problem into several smaller problems that are inputs to the analog signed multiplier blocks. The output of the analog block is a thermometer code that is resolved and combined with the results of the other analog multipliers in the logic blocks at the right of the figure.

Each *L*-bit multiplicand (*A* and *B*) is divided into *L/N* *N*-bit portions. Each portion of *A* is multiplied by each portion of *B*, so that the number of analog signed multipliers, *M*, is

$$M = \left(\left\lceil \frac{L}{N} \right\rceil \right)^2 \tag{1}$$

2.4. Analog Multiplier Implementation

The analog signed multipliers each consist of a Gilbert cell multiplier and supporting DACs that produce a differential current, $I_1 - I_2$, representing the signed multiplication of two *N*-bit values. An offset current, I_{off}, is added to the differential current to make the current into the subsequent integrate-and-dump circuit unipolar. The integration time is denoted T_{int}. The result of the integration, a voltage that ranges from 0 volts to V_{ref}, is passed to several comparators to determine the final multiplication result. The number of comparators is a function of the size of the multiplier and is referred to as $f_c(N)$ in the equations below. The minimum voltage size difference between output levels is ΔV, which appears prominently in the signal-to-noise calculations.

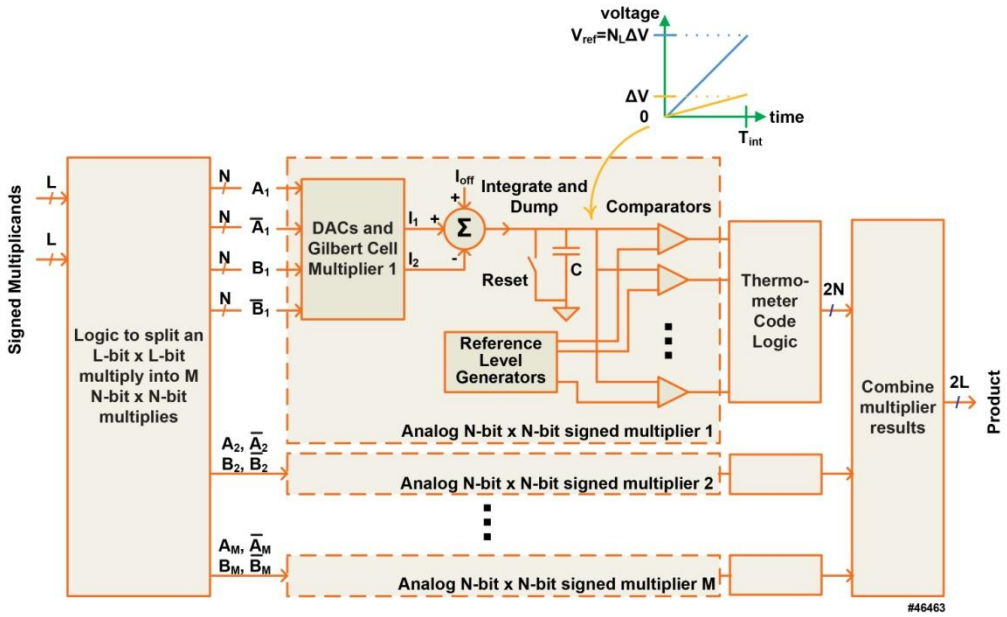

Figure 1. Analog Implementation of an L-bit x L-bit Signed Multiplier

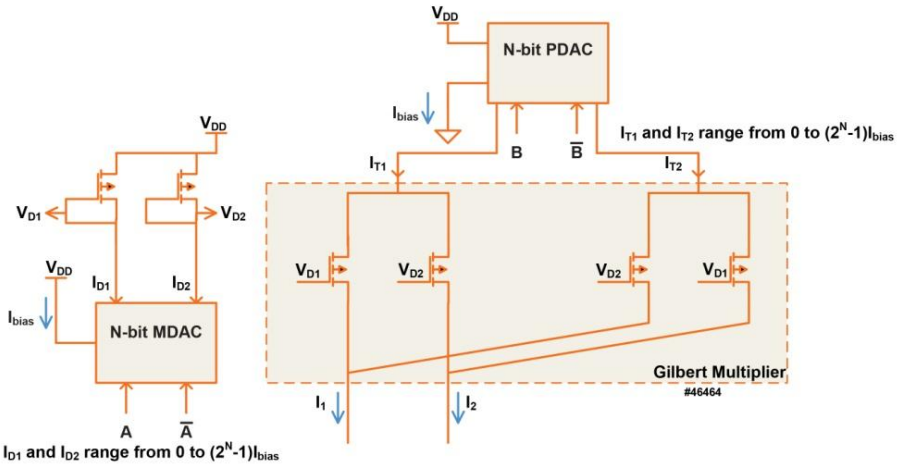

Figure 2. Gilbert Cell Multiplier with DACs

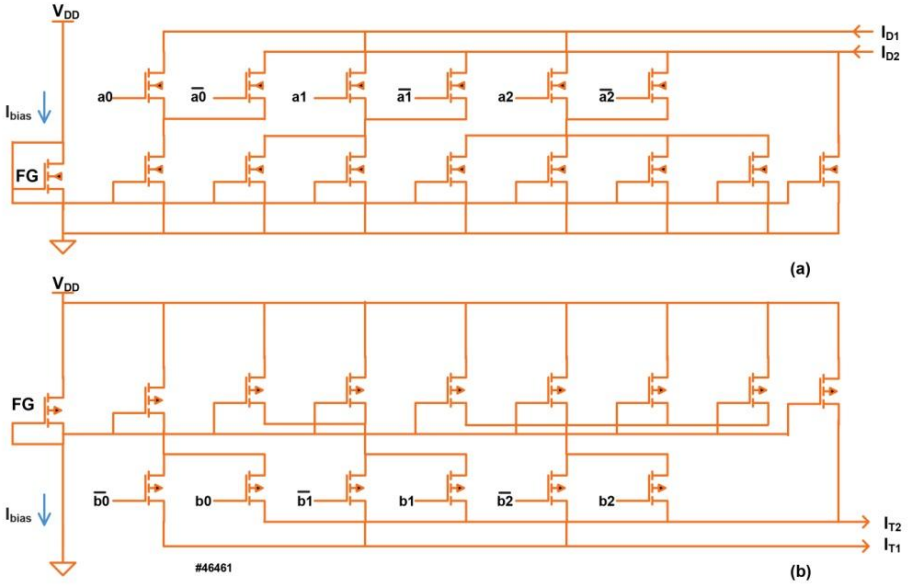

Figure 3. Example of a 3-bit MDAC and PDAC using Floating Gate Transistors to Set the Bias Current

The Gilbert cell multiplier block, including DACs at the front end, is shown in Figure 2. A Gilbert cell was chosen to implement the multiplication because transistors in their sub-threshold region have characteristics that give a very linear response to current inputs to the cell [7]:

$$I_1 - I_2 = \frac{(I_{T1} - I_{T2})(I_{D1} - I_{D2})}{I_{D1} + I_{D2}} \quad (2)$$

The PDAC ("positive DAC") generates I_{T1} and I_{T2} based on the N-bit digital value B, while the MDAC ("minus DAC") generates I_{D1} and I_{D2} based on the N-bit digital value A.

The DACs take a reference current that is controlled by a floating-gate FET and steer multiples of that current to the output pins of the DAC. Figure 3 shows an example of a 3-bit MDAC (a) and a 3-bit PDAC (b).

The two outputs of the Gilbert multiplier, I_1 and I_2, are subtracted and an offset, I_{off}, is added to them to produce a unipolar current I_s that drives the integrate-and-dump circuit (Figure 4).

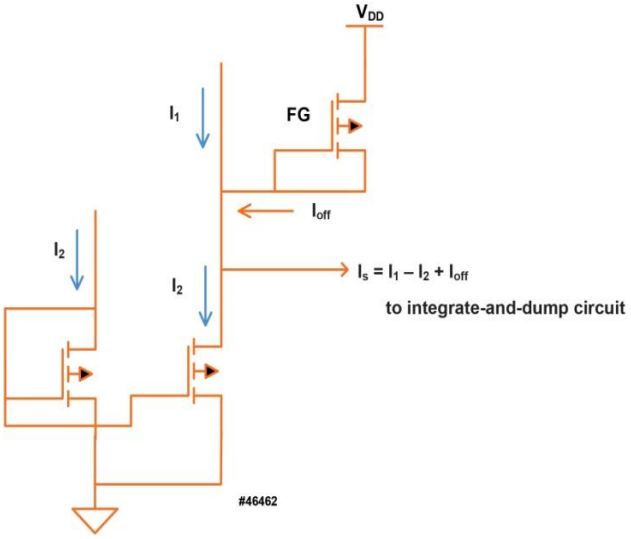

Figure 4. Circuitry Creating Unipolar Current into the Integrate-and-Dump

The I_s current is passed through an integrator ("integrate and dump") with an integration capacitor C (Figure 1) giving the voltage at the comparators as:

$$v_{out} = \frac{1}{C}\int_{t_0}^{t_0+T_{int}} I_s dt \tag{3}$$

The signed multiplier has some interesting characteristics that should be noted, as summarized in Table 1. For example, with a 2-bit x 2-bit multiplier, the possible digital inputs in 2's complement representation are {-2, -1, 0, 1}. The multiplication result of two numbers is one of {-2, -1, 0, 1, 2, 4}. The number of comparators needed to differentiate between these values is 5. The span of values is 4 – (-2) = 6, and so $\Delta V = V_{ref}/6$.

Table 1. Analog Multiplier Characteristics

Parameter	Symbol	Value
DAC Outputs 1	I_{T1}, I_{D1}	0 to $(2^N\text{-}1)I_{bias}$
DAC Outputs 2	I_{T2}, I_{D2}	I_{bias} to $2^N I_{bias}$
DAC Output Sums	$I_{T1}+I_{T2}$, $I_{D1}+I_{D2}$	$2^N I_{bias}$
Multiplier Output Range	$I_1 - I_2$	$\{2^{N-1}\text{-}2^{2N-2}$ to $2^{2N-2}\}I_{bias}$
Multiplier Output Sum	$I_1 + I_2$	$2^N I_{bias}$
Added Offset Current	I_{off}	$(2^{2N-2}\text{-}2^{N-1})I_{bias}$
Level Separation	ΔV	$V_{ref}/(2^{2N-1} - 2^{N-1})$
Number of Comparators	$f_c(N)$	$f_c(2)=5$; $f_c(3)=17$; $f_c(4)=60$

2.5. Analog Multiplier Energy Consumption

The static power consumption of the N-bit x N-bit analog multiplier is the power supply voltage multiplied by the current that it draws from the supply. The supply currents include (1) the MDAC and PDAC currents, (2) the added offset current, (3) the currents from the reference level generators, and (4) the currents from the comparators.

$$\frac{P_{static}}{V_{DD}I_{bias}} = 2\left(2^N+1\right)+\left(2^{2N-2}-2^{N-1}\right)+3f_C(N) \tag{4}$$

The total energy for all M multipliers (implementing the full L-bit x L-bit multiply circuit) is the power multiplied by the integration time T_{int} (using (1)):

$$\frac{E_{static}}{V_{DD}I_{bias}T_{int}} = \left(\left\lceil \frac{L}{N} \right\rceil\right)^2\left[\begin{array}{c} 2\left(2^N+1\right)+ \\ \left(2^{2N-2}-2^{N-1}\right)+3f_C(N) \end{array}\right] \tag{5}$$

While the power supply voltage, V_{DD}, is set by the technology, the integration time is determined by the signal-to-noise ratio at the output of the analog multipliers.

2.6. Signal to Noise Ratio Analysis

The output of the integrate-and-dump circuit feeds comparators with a minimum difference between levels of ΔV. That means that a signal deviation from the ideal levels of $\Delta V/2$ will cause an error in the multiplication, so we define the SNR voltage ratio as (where ΔV is defined in Table 1):

$$SNR = \frac{(\Delta V/2)}{RMS\ noise} \tag{6}$$

This SNR can be used to determine the error rate, P_e, of the N-bit x N-bit multiplier:

$$P_e = Q(SNR); \quad Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty \exp\left(-u^2/2\right)du \tag{7}$$

Our noise analysis will include thermal and shot noise. Flicker (*1/f*) noise is not included so that we can determine the best obtainable performance if flicker noise is reduced as described in section 2.7. We are guided by [8] in calculating the noise due to currents in the FETs. To determine the noise, we start with I_s, the current into the integrate-and-dump circuit. The noise is the sum of the noise components due to the three currents entering the summing node of Figure 4. Since $I_{sum} = I_1 + I_2 + I_{off}$ is a constant (Table 1), then the white noise generated by the current is also constant, and the noise density (using q as the elementary charge) is:

$$N_{0,Is} = 2qI_{sum} \quad [A^2/Hz] \tag{8}$$

where:

$$I_{sum} = I_{bias}\left(2^N + 2^{2N-2} - 2^{N-1}\right) \tag{9}$$

This noise current passes through the integrator described by (3), which is equivalent to a convolution of the noise current with a rectangular filter $h(t)$ over the integration time:

$$h(t) = \begin{cases} \frac{1}{C}, 0 \le t \le T_{int} \\ 0, \quad otherwise \end{cases} \tag{10}$$

Let $H(f)$ be the Fourier transform of $h(t)$. The noise power at the output of the integrator (using Parseval's theorem) is:

$$\begin{aligned} \sigma_1^2 &= \frac{N_{0,Is}}{2}\int_{-\infty}^{\infty}|H(f)|^2 df \\ &= \frac{N_{0,Is}}{2}\int_{-\infty}^{\infty}h(t)^2 dt = \frac{qI_{sum}T_{int}}{C^2} \end{aligned} \tag{11}$$

This representation can be simplified with the realization that the maximum integrator voltage, V_{ref}, is reached with the maximum integrator output current, which is I_{sum}. That means:

$$V_{ref} = \frac{I_{sum}T_{int}}{C} \tag{12}$$

$$\sigma_1^2 = \frac{qV_{ref}}{C} \tag{13}$$

There is also thermal noise due to the integration capacitor, C, where k is the Boltzmann constant and T is the temperature:

$$\sigma_2^2 = \frac{kT}{C} \tag{14}$$

The total noise is the sum of σ_1^2 and σ_2^2. Recognizing, however, that $qV_{ref} >> kT$, the output noise variance reduces to σ_1^2, and the RMS noise is just the square root of (13).

Using the values for ΔV from Table 1, I_{sum} from (9), the RMS noise from (13), and V_{ref} from (12), the signal-to-noise ratio of (6) can be expressed as:

$$SNR = \frac{\sqrt{\left(2^N + 2^{2N-2} - 2^{N-1}\right) I_{bias} T_{int}}}{2\left(2^{2N-1} - 2^{N-1}\right)\sqrt{q}} \quad [V/V] \quad (15)$$

This equation can be rearranged to solve for $I_{bias}T_{int}$ and then substituted into (5), yielding the static energy for the L-bit by L-bit multiply as a function of the SNR.

$$E_{static} = V_{DD}\left(\left\lceil \frac{L}{N} \right\rceil\right)^2 \left[2\left(2^N + 1\right) + \left(2^{2N-2} - 2^{N-1}\right) + 3 f_C(N) \right] \bullet \left[\frac{SNR^2 \left[2\left(2^{2N-1} - 2^{N-1}\right)\right]^2 q}{\left(2^N + 2^{2N-2} - 2^{N-1}\right)} \right] \quad (16)$$

E_{static} is the energy consumed per L-bit x L-bit multiply with a specified error rate, P_e, which determines the required SNR using (7) for the individual N-bit x N-bit multiplies, and does not include any energy of the digital circuitry leading into or out of the analog circuitry.

The error rate of the entire L-bit x L-bit multiply circuit, P_{err}, is given by:

$$P_{err} = 1 - (1 - P_e)^M \approx MP_e \quad (17)$$

That is to say, the error rate of the individual N-bit x N-bit multipliers must be M times better than the desired overall error rate.

2.7. Flicker Noise Considerations

The signal-to-noise calculations in section 2.6 neglect the effects of flicker noise, which can be a significant noise contributor in CMOS circuits. Because the magnitude of flicker noise depends on the CMOS transistor geometry (flicker noise reduces as channel length and widths are increased [17]), the amount of flicker noise could in principle be reduced to be less than the thermal and shot noises if the area penalty were acceptable. Thermal and shot noise terms cannot be reduced by changing transistor geometries and so the ultimate circuit performance is limited by the SNR determined in section 2.6.

2.8. Static Energy Calculation Results

Figure 5 shows both the static energy required for 8-bit x 8-bit and 16-bit x 16-bit multiplies using smaller analog multipliers with N=2, 3, and 4 as a function of operation error rate as well as the energy with N=2 (the most promising data) under the assumption that V_{DD}=0.7 volts, which is reasonable for a state-of-the-art technology.

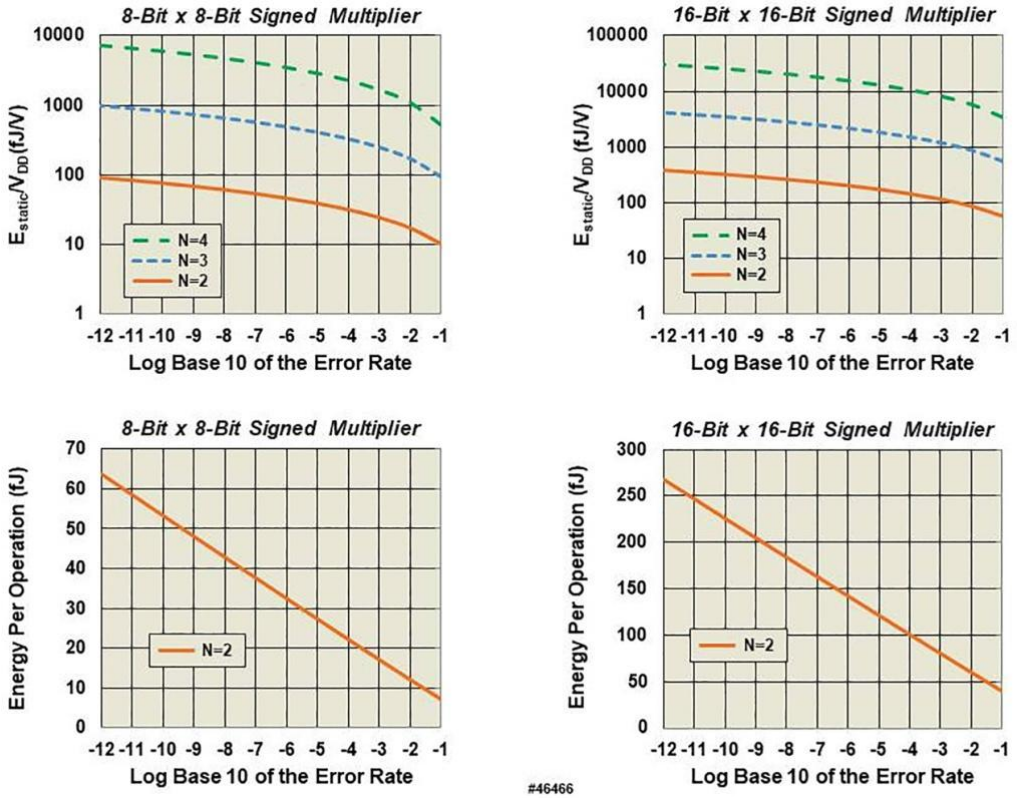

Figure 5. Normalized Analog Energy and Energy Consumption at V_{DD}=0.7 volts (Note: as expected, $E_{16x16} \approx 4*E_{8x8}$)

3. Digital Implementation Comparison

3.1. Digital Implementation

In this portion of the paper we estimate the static and dynamic power and energy consumption of digital multipliers implemented in a state-of-the-art technology to act as a benchmark with which to compare the energy consumption results of the analog multipliers. We concentrate on the characteristics of two signed multipliers: an 8-bit x 8-bit (producing a 16-bit result) and a 16-bit x 16-bit (producing a 32-bit result).

The methodology was to write a high-level description of the multipliers in VHDL and allow a synthesis tool [9] to create a "vanilla" implementation of the two multipliers using a state-of-the-art standard-cell library. No special care was taken to optimize the multipliers for power. The library included half- and full-adder cells, and the implementation took advantage of these (Figure 6, Table 2).

Table 2. Standard Cell Implementation Statistics

Design	Number of Wires	Number of Standard Cells (Excluding Latches)	Number of Standard Cells (Including Latches)	Cell Area (16 nm Technology) μm^2 (Including Latches)
8b x 8b	160	146	178	92
16b x 16b	593	499	563	293

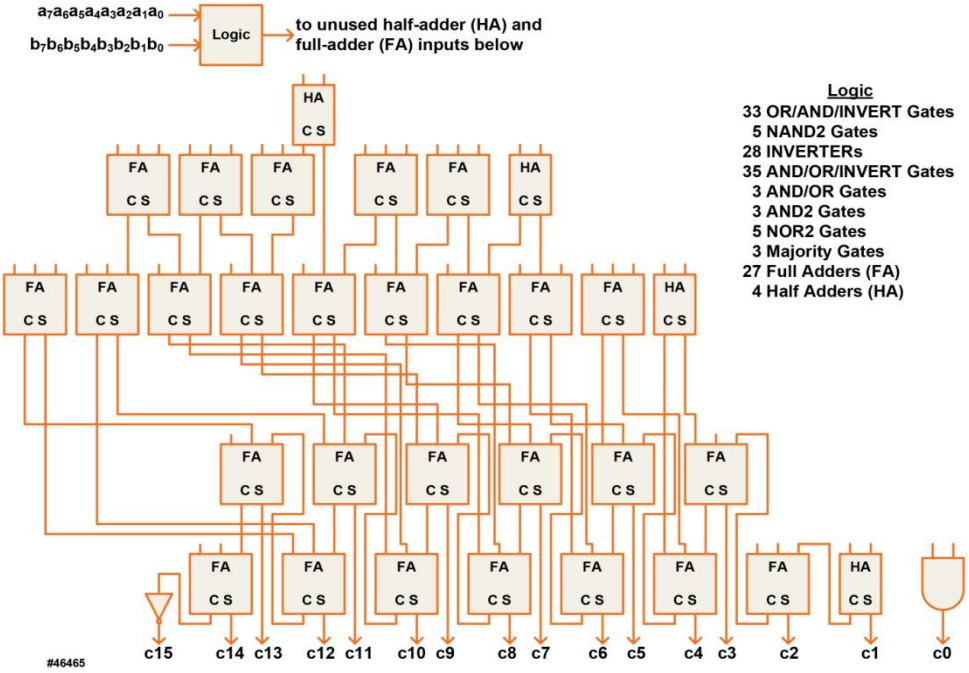

Figure 6. Implementation of a Digital 8-bit x 8-bit Signed Multiplier

3.2. Dynamic Power Estimation

Active CMOS energy consumption can be divided into two portions: (1) the energy consumption of nets (traces) between standard cells and (2) the energy consumption of the standard cells themselves, including energy to charge any FETs connected to input and output nets. The energy consumption depends on the technology characteristics listed in Table 3, which includes assumed values for a state-of-the-art (e.g., 7 nm) technology.

The average net length was determined by first using data from [10], which estimates that an average net in a design is 40% of the length of the side of square layout for approximately the number of cells in our designs. Applying a scaling factor of ¼ to the area when going from 16 nm to 7 nm [11] gives the values in Table 3.

Table 3. Technology Parameter Definitions

Technology Parameter	Definition	Process Value
L	Average length of nets in multiplier (µm). Unique to each multiplier.	L_{8x8} = 1.9 µm L_{16x16} = 3.4 µm
c	Capacitance per length of wire (fF/µm)	c = 0.4 fF/µm
V_{DD}	Supply voltage	V_{DD} = 0.7 V
C_{ggn}	NFET gate capacitance	C_{ggn} = 40 aF
C_{ggp}	PFET gate capacitance	C_{ggp} = 40 aF
C_{dsn}	NFET drain-to-source capacitance	C_{dsn} = 20 aF
C_{dsp}	PFET drain-to-source capacitance	C_{dsp} = 20 aF

Estimates of trace capacitance for a 7 nm process range from 0.35 fF/µm [12] to 0.49 fF/µm [13]; we used 0.4 fF/µm.

[11] indicates a 7 nm process supply voltage will range from 0.5 volts to 0.7 volts. [14] asserts 0.7 volts for the processes that it knows about. We selected 0.7 volts.

Values of the gate parasitic capacitances and drain-to-source parasitic capacitances of NFETs and PFETs were taken from plots in [15] for a 10 nm FinFET technology.

Simulations were run of the two multipliers, noting the activities of both the nets connecting the standard cells as well as the internal nodes of the standard cells. Energy is drawn from the supplies only on upward transitions of the nets. Let *n* be the number of nets and the other variables as in Table 3,

$$a = \frac{total\ number\ of\ 0 \rightarrow 1\ transitions}{(number\ of\ nets)(number\ of\ clocks)} \tag{18}$$

$$E_{wires} = anLcV_{DD}{}^2 \tag{19}$$

Activity factors of 23.3% and 23.7% were respectively observed for the 8-bit x 8-bit and 16-bit x 16-bit multipliers.

Similarly, activity factors were calculated for 0→1 transitions on the gates and drains of the standard cell transistors (N_{cggn} is the number of upward transitions on C_{ggn}, etc.):

$$E_{stdcells} = \frac{V_{DD}{}^2}{number\ of\ clocks}\left[N_{Cggn}C_{ggn} + N_{Cggp}C_{ggp} + N_{Cdsn}C_{dsn} + N_{Cdsp}C_{dsp}\right] \tag{20}$$

$$E_{active\_per\_op} = E_{wires} + E_{stdcells} \tag{21}$$

The results of the simulation are in Table 4.

Table 4. Active Energy Estimates at 7 nm

Multiplier	E_{wires}: energy due to the wires, equation (19)	$E_{stdcells}$: energy due to the standard cells, equation (20)	$E_{active\_per\_op}$: total energy, equation (21)
8b x 8b	13.9 fJ	9.2 fJ	23.1 fJ
16b x 16b	93.8 fJ	36.6 fJ	130.4 fJ

3.3. Static Power Estimation

Due to leakage current in the FET devices, CMOS circuits consume power even when not switching, which is called static power. [16] estimates a 16-bit x 16-bit multiplier operating at a supply voltage of 0.45 volts to have a leakage power of 1.82 microwatts. Scaling the power supply voltage to 0.7 volts gives 2.8 microwatts of static power.

Since leakage power is proportional to the number of transistors in a circuit, we estimate the 8-bit x 8-bit multiplier power as 178/563 (Table II) of the 16-bit x 16-bit multiplier, or 0.9 microwatts.

3.4. Total Energy per Operation Estimation

The total energy per operation depends on the clock speed *f:*

$$E_{per\_op} = \frac{P_{static}}{f} + E_{active\_per\_op} \tag{22}$$

The results are shown in Figure 7.

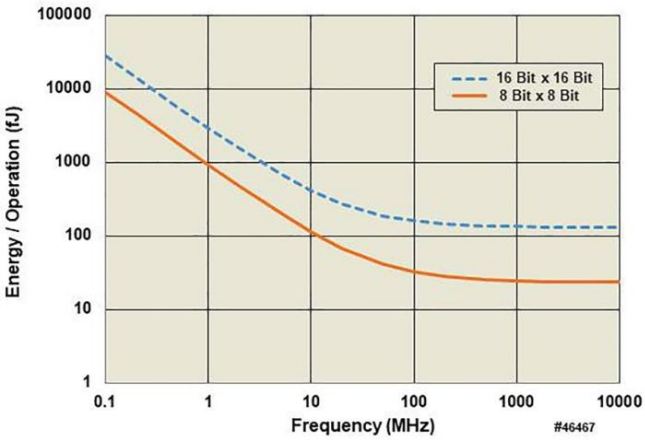

Figure 7. Total Energy per Operation for Digital Signed Multipliers

4. Discussion and Conclusions

For the digital implementation of the 8-bit x 8-bit and 16-bit x 16-bit multipliers, Figure 7 indicates that the energy consumption per multiplication operation is dominated by the dynamic energy consumption for operating frequencies greater than 100 megahertz, which is certainly reasonable. That means that the energy consumption benchmark is the "$E_{active\_per\_op}$" column of Table 4.

A comparison of these energy consumption values against Figure 5 shows that **the analog circuitry energy consumption is higher than the equivalent digital circuit energy consumption unless the system error rate is allowed to be about 1 error in 10^5 multiply operations.**

The analog energy consumption is independent of the bias current, and is instead proportional to the bias current multiplied by the integration time. This may not be surprising, since the product of the bias current, the power supply voltage, and the integration time has units of energy.

The analog circuit energy consumption was determined by and was directly related to the square of the (voltage/voltage) signal-to-noise ratio, and so it quickly increased as the number of bits in the analog multiplier was increased. The most efficient decomposition of the large multiplier was to use the smallest analog multiplier.

The analysis of the analog multiplier was optimistic; it assumed that flicker noise could be reduced to a negligible level through transistor scaling and did not consider the energy consumption of additional circuitry that might be needed to compensate for temperature drift. The analysis also did not include the energy consumption of the digital logic surrounding the analog computation core.

It may be that the Gilbert cell multiplier is not the most power efficient analog multiplier. We briefly considered a multiplying DAC implementation, but it appeared to consume more power than the Gilbert cell implementation.

REFERENCES

[1] S. George, S. Kim, S. Shah, J. Hasler, M. Collins, F. Adil, R. Wunderlich, S. Nease, and S. Ramakrishnan, "A programmable and configurable mixed-mode FPAA SoC," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 24, no. 6, pp. 2253-2261, 2016.

[2] G. Cowan, R. Melville and Y. Tsividis, "A VLSI analog computer/digital computer accelerator," IEEE Journal of Solid-State Circuits, vol. 41, no. 1, pp. 42-53, 2006.

[3] R. Chanpuriya, A. Shrivastava and V. Magraiya, "Process aware circuit design using adaptive body biasing," International Journal of Engineering and Innovative Technology, vol. 2, no. 12, pp. 10-14, 2013.

[4] A. Aslam-Siddiqi, W. Brockherde, M. Schanz and B. Hosticka, "A 128-pixel CMOS image sensor with integrated analog nonvolatile memory," IEEE Journal of Solid-State Circuits, vol. 33, no. 10, pp. 1497-1501, 1998.

[5] V. Srinivasan, D. Graham and P. Hasler, "Floating-gates transistors for precision analog circuit design: An Overview," in 48th Midwest Symposium on Circuits and Systems, 2005.

[6] N. Beyraghi and A. Khoei, "CMOS eesign of a low power and high precision four-quadrant analog multiplier," International Journal of Electronics and Communications, vol. 69, pp. 400-407, 2015.

[7] C. Remund, "Design of CMOS four-quadrant Gilbert cell multiplier circuits and weak and moderate inversion (MS Thesis)," 24 November 2004. [Online]. Available: https://scholarsarchive.byu.edu/cgi/viewcontent.cgi?article=1207&context=etd. [Accessed 24 August 2018].

[8] R. Sarpeshkar, T. Delbruck, C. Mead, "White noise in MOS transistors and resistors," IEEE Circuits and Devices Magazine, vol.9, no. 6, pp. 23-29, 1993.

[9] Cadence Design Systems, Inc., "Genus synthesis solution," 2019. [Online]. Available: https://www.cadence.com/content/cadence-www/global/en_US/home/tools/digital-design-and-signoff/synthesis/genus-synthesis-solution.html. [Accessed 24 Jan 2019].

[10] M. Pedram, "Interconnection length estimation for Optimized standard cell layouts," in IEEE International Conference on Computer-Aided Design, Santa Clara, CA, 1989.

[11] S. Jones, "14nm 16nm 10nm and 7nm - what we know now," 7 Apr 2017. [Online]. Available: https://www.semiwiki.com/forum/content/6713-14nm-16nm-10nm-7nm-what-we-know-now.html. [Accessed 14 Nov 2018].

[12] I. Ciofi, A. Contino, P. Roussel, R. Baert, V.-H. Vega-Gonzalez, K. Croes, M. Badaroglu, C. Wilson, P. Raghavan, A. Mercha, D. Verkest, G. Groeseneken, D. Mocuta and A. Thean, "Impact of wire geometry on interconnect RC and circuit delay," IEEE Transactions on Electron Devices, vol. 63, no. 6, pp. 2488-2496, 2016.

[13] S. Sinha, B. Cline, G. Yeric and V. Chandra, "Design genchmarking to 7nm with FinFET predictive technology models," in Proceedings of the 2012 ACM/IEEE International Symposium on Low Power Electronics and Design, Redondo Beach, CA, USA, 2012.

[14] WikiChip LLC, "7 nm lithography process," 28 Oct 2018. [Online]. Available: https://en.wikichip.org/wiki/7_nm_lithography_process. [Accessed 14 Nov 2018].

[15] N. Fasarakis, A. Tsormpatzoglou, D. Tassis, I. Pappas, K. Papathanasiou, M. Bucher, G. Ghibaudo and C. Dimitradis, "Compact capacitance model of undoped or lightly doped ultra-scaled triple-gate FinFETs," IEEE Transactions on Electron Devices, vol. 59, no. 12, pp. 3306-3312, 2012.

[16] Q. Xie, X. Lin, Y. Wang, S. Chen, M. J. Dousti and M. Pedream, "Performance comparisons between 7nm FinFET and conventional bulk CMOS standard cell libraries," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 62.

[17] H. Tuinhout and A. Zegers-van Duijnhoven, "Evaluation of 1/f noise variability in the subthreshold region of MOSFETs," 2013 IEEE International Conference on Microelectronic Test Structures, Osaka, Japan, 2013.