Implementing Digital Logic with Sinusoidal Supplies

Kalyana C. Bollapalli, Sunil P. Khatri, Laszlo B. Kish
Department of ECE, Texas A&M University, College Station TX 77843

***Abstract*— In this paper, a new type of combinational logic circuit realization is presented. Logic values are implemented as sinusoidal signals. Sinusoidal signals of the same frequency are phase shifted by π to destructively interfere with each other, and represent the logic 0 and 1 values of Boolean Logic. These properties of sinusoids can be used to identify a signal without ambiguity. Thus, representing logic values as sinusoidal signals yields a realizable system of logic. The paper presents a logic gate family that can operate using the sinusoidal signals for logic 0 and logic 1 values. Due to orthogonality of sinusoid signals with different frequencies, multiple sinusoids could be transmitted on a single wire. This provides a natural way of implementing multilevel logic. Signals traveling long distances could take advantage of this fact and can share interconnect lines. Recent research in circuit design has made it possible to harvest sinusoidal signals of the same frequency and 180° phase difference from a single resonant clock ring in a distributed manner. Other advantage of such a logic family is its immunity from external additive noise. The experiments in this paper indicate that this paradigm, when used to implement binary valued logic, yields an improvement in switching (dynamic) power.**

I. Introduction

With the recent slowing of Moore's law, unconventional ways to perform computation are being investigated. Signals that are not DC in nature are being used to represent logic values, and their properties are being used for deterministic computation. An example of a signal that is not DC in nature and can be used for deterministic computation is thermal noise generated by resistors [1], [2], [3]. In this paper we explore sinusoid signals to represent logic values. One of the properties of sinusoid signals is that signals with different frequencies do not have any correlation. This property ensures that only signals with the same frequency can correlate, and hence can be used to deterministically identify signals. The aforesaid property of sinusoid signals could be exploited to implement different logic values using different frequencies and different phases. Let S_i^k represent the sinusoid of frequency i and phase k and let $<,>$ represents the correlation operator, then $\delta_{i,j}^{k,l} :=< S_i^k, S_j^l >; i \neq j \rightarrow \delta_{i,j}^{k,l} = 0; i = j \rightarrow \delta_{i,j}^{k,l} = cos(k-l)$. In this paper we use the above mentioned ideas and present a way to realize logic gates that can compute with sinusoidal supplies. We also explore design considerations that are induced due to the nature of the supplies and the methodology of computation. Logic gates using sinusoid supplies present us with some attractive advantages

- *Additive noise immunity*: Additive noise does not affect the orthogonality of logic values, as opposed to the traditional representation of logic values, where signal integrity degrades with additive noise.
- *Multivalued logic*: Sinusoids with different frequencies do not correlate. Multivalued logic can be realized by using sinusoidal signals of different frequencies. Wherein different logic values can be represented by different frequencies or as a superimposition of multiple frequencies. Multivalued logic could be used to reduce the logic depth and thereby reduce circuit area.
- *Interconnect Reduction*: The above mentioned property of sinusoids (non-correlation of sinusoids of different frequencies) could also be used for multiple signals to share a single wire. This could be exploited by signals that have to be routed over large distances, thereby reducing wiring area.

The contributions of this paper are

- Demonstrating that logic gates can be implemented with sinusoidal supply signals.
- Exploring design considerations induced by the methodology of computation, and quantifying the induced design parameters.

The remainder of this paper is organized as follows. Section II discusses some previous work in this area. In Section III we describe our method to implement logic gates using sinusoidal supplies. Section IV presents experimental results, while conclusions are discussed in Section V.

II. Previous Work

There has been significant amount of research in conventional ways of representing logic. Recently, newer ways to represent logic values have been envisioned. In [4], authors have proposed using independent stochastic vectors for noise robust information processing. In their approach, an input signal is compared with a stored reference signal using a comparator. The number of rising edges in the output of the comparator is computed. The authors claim that as the similarity between signals increases the number of rising edges counted would also increase. The authors present results averaged over a large number of simulations, and show that similar signals, on an average, result in larger count values. Their work presents a way to correlate two signals, but their implementation requires complex hardware units (a comparator and a counter). Also, their method can not deterministically identify a signal and therefore can not be used to implement logic gates. In [1], [2], authors have proposed using noise for deterministic logic and digital computing. Independent noises are also uncorrelated (orthogonal) stochastic processes. The author presents an approach to implement noise-based logic gates. In a binary logic system, there would be two reference noise signals (one signal per logic value) with which the input is correlated. A thermal noise signal can be identified by correlating it with all the reference noise signals. A correlator could be realized as an analog multiplier followed by a time averaging unit like a low pass filter. The result of the time averaging unit can then be used to drive the output of the gate using either the logic 0 noise or logic 1 noise value or their superimposition. However, there are many open problems and questions with utilizing the noise. Calculations indicate a factor of 500 slowdown compared to the noise bandwidth otherwise error rates are higher than normal. Significant power reduction can only be expected with small noise and that needs preamplifier stages in the logic gates which needs extra power. In [5], authors build a multidimensional logic hyperspace with properties similar to quantum Hilbert space. They provide a noise-based deterministic string search algorithm which outperforms Grover's quantum search engine [5]. In this paper, we seek an alternative way to implement logic. Among the features we desire are a) representation in which the duration for computing correlation can be deterministically predicted, and b) a representation that allows us to choose sources that are not very sensitive to load. In the next section we describe our approach to realizing such a logic family, which uses some starting ideas from [2].

III. Our Approach

We describe our approach with an example. Figure 1 illustrates the circuit diagram of a basic gate. This gate consists of two mixers, two integrators (low pass filters) and a output driver.

The input signal (*in*) is mixed (multiplied) with reference sinusoidal signals (S_0 and S_1) which represent the logic 0 and logic 1 values. Equation 1 describes the result of mixing two sinusoidal signals.

$$\begin{aligned} sin(\omega_1 t + \delta_1) \otimes sin(\omega_2 t + \delta_2) = \\ \tfrac{1}{2} \times (cos((\omega_1 - \omega_2)t + (\delta_1 - \delta_2)) - \\ cos((\omega_1 + \omega_2)t + (\delta_1 + \delta_2))) \end{aligned} \quad (1)$$

If $\omega_1 = \omega_2$, the low frequency term of Equation 1 ($cos((\omega_1 - \omega_2)t + (\delta_1 - \delta_2))$) would result in a constant. The output of the mixer is integrated using a low-pass filter (LPF). The LPF filters out the high frequency component of Equation 1. If the signals that are being

978-3-9810801-6-2/DATE10 © 2010 EDAA

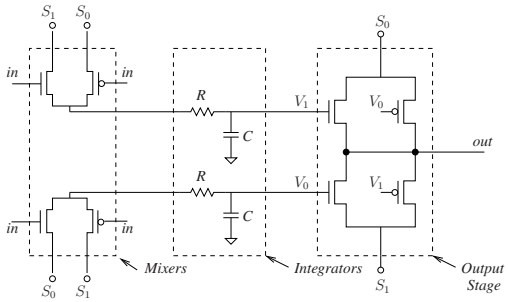

Fig. 1. Logic Gate

mixed are in phase ($\delta_1 = \delta_2$) the result of the integrator is a positive constant. If the signals are out of phase ($\delta_1 = \delta_2 \pm \pi$) the result of integration is a negative constant. Thus if sinusoidal sources S_0 and S_1 are of the same frequency and are out of phase by 180°, they would least correlate. We now describe a way of practically realizing two signals of the same frequency, and a 180°, phase difference.

A. Standing wave oscillator

Recent research in the field of low power high frequency clock generation has shown the feasibility of realizing a distributed LC oscillator on chip [6][7][8]. These oscillators use the parasitic inductance of on-chip interconnect, along with the parasitic capacitance presented by interconnect as well as semiconductor devices to realize a high frequency resonant oscillator. Such oscillators include a negative resistance structure to compensate for the resistive loses in the interconnect. In this work, we use a standing wave resonant oscillator from [8].

B. A Basic Gate

We now describe a basic inverter gate. To be able to use these gates in large integrated circuits, each of the components of the gate has to be implemented with as few devices as possible. Figure 1 illustrates the structure of a inverter in our implementation. The structure of any sinusoidal supply based logic gate would be very similar to the gate in Figure 1. The input of the gate is a sinusoid signal. The input signal is correlated with all the reference sinusoidal sources (2 sources in this instance). Based on the logic value identified on the wire, and based on the logic implemented in the gate, a corresponding sinusoidal signal is driven out. A combination of a mixer and an integrator is used as a correlation operator. The output of the integrator is then used to enable pass gates which drive out the appropriate reference sinusoidal signal.

1) Multiplier: To implement a multiplier with two devices, we bias an NMOS device and a PMOS device in the linear region. Equation 2 presents the formula for the drain current in an NMOS when the device in is linear region. The drain current is dependent on a product of gate voltage (V_{gs}) and drain voltage (V_{ds}). The drain current can thus be used to extract the result of multiplication.

$$I_{ds}(t) = \beta(V_{gs}(t) - V_{th} - \frac{V_{ds}(t)}{2})V_{ds}(t) \tag{2}$$

$$I_{ds}(t) = \beta(V_{gs}(t)V_{ds}(t) - V_{th}V_{ds}(t) - \frac{{V_{ds}(t)}^2}{2})$$

If this drain current is driven through a suitable low pass filter to block the high frequency components, we can extract the average drain current. Let $\overline{I}$ indicate the average value of $I(t)$.

$$\overline{I_{ds}} = \beta(\overline{V_{gs}V_{ds}} - \overline{V_{th}V_{ds}} - \frac{\overline{V_{ds}^2}}{2})$$

Let $V_{gs}(t)$ and $V_{ds}(t)$ be signals of the same frequency and a phase difference of α. Without loss of generality, let $V_{gs}(t) = V_1 sin(\omega t)$ and $V_{ds} = V_1 sin(\omega t + \alpha)$. Then from Equation 1,

$$\overline{V_{gs}V_{ds}} = \frac{cos(\alpha)}{2}V_1^2$$

and

$$\overline{V_{ds}V_{ds}} = \frac{1}{2}V_1^2$$

Also since V_{th} is a constant, $V_{th}V_{ds}(t)$ is a high frequency sinusoidal signal, and thus $\overline{V_{th}V_{ds}} = 0$. Hence,

$$\overline{I_{ds}} = \beta(\frac{cos(\alpha)}{2}V_1^2 - 0 - \frac{1}{4}V_1^2)$$

$$\overline{I_{ds}} = \left(\frac{2cos(\alpha) - 1}{4}\right)\beta V_1^2 \tag{3}$$

If the signals $V_{gs}(t)$ and $V_{ds}(t)$ are signals of the same frequency with $\alpha = 0$, $\overline{I_{ds}}$ is positive. If they are of same frequency with $\alpha = \pi$, $\overline{I_{ds}}$ is negative. The above result shows that when signals correlate, the result is a average positive drain current, but when the signals do not correlate there is an average negative drain current.

A similar effect can be achieved with a PMOS device. The drain current of a PMOS transistor in linear region is given by

$$I_{sd}(t) = -\beta(V_{gs}(t) - V_{th} - \frac{V_{ds}(t)}{2})V_{ds}(t) \tag{4}$$

By the same argument as above, if V_{gs} and V_{ds} are sinusoidal signals of the same frequency, same amplitude and differ in phase by α,

$$\overline{I_{sd}} = -\left(\frac{2cos(\alpha) - 1}{4}\right)\beta V_1^2 = \left(\frac{1 - 2cos(\alpha)}{4}\right)\beta V_1^2 \tag{5}$$

If the signals V_{gs} and V_{ds} were in phase ($\alpha = 0$), $\overline{I_{sd}}$ is negative. But if $\alpha = \pi$, $\overline{I_{sd}}$ is positive.

From Equations 3 and 5 we can conclude that the drain current is positive when the NMOS device mixes signals that are in phase, and is positive when PMOS mixes signals that are out of phase. Thus an NMOS device mixing the input with one reference sinusoidal signal and PMOS mixing the same input with the other reference sinusoidal signal forms a good *complimentary* mixer. Figure 1 shows the circuit diagram of the above described mixer. The total average drain current that can be driven by such a mixer is (from Equation 3 and 5)

$$\overline{I} = \left(\frac{2cos(\alpha) - 1}{4} + \frac{1 - 2cos(\pi - \alpha)}{4}\right)\beta V_1^2$$

$$\overline{I} = \left(\frac{2cos(\alpha) - 1}{4} + \frac{1 + 2cos(\alpha)}{4}\right)\beta V_1^2$$

$$\overline{I} = cos(\alpha)\beta V_1^2 \tag{6}$$

2) Low Pass Filter: The choice of R and C values of the integrator poses an optimization problem. A small RC product ensures that the voltage on the capacitor reaches the final DC value early, but a large fraction of the sinusoid will be visible at output node. On the other hand a large value of RC product ensures that a smaller input signal is visible on the capacitor but the DC voltage on the capacitor would take longer to reach its final DC value. We have:

$$\frac{1}{RC} = 3F$$

where F is the corner frequency of the low pass filter. From the above equation we can compute the magnitude of input signal that is visible on the output of the low pass filter.

$$\frac{\frac{1}{j\omega C}}{R + \frac{1}{j\omega C}} = \frac{1}{j\omega RC + 1}$$

$$= \frac{1}{\frac{j\omega}{3F} + 1} \tag{7}$$

Based on the input frequency, a low pass filter can be designed to minimize the peak to peak amplitude of input signal visible after the correlator. The resistance is realized using resistive poly-silicon, and the capacitance is realized as gate capacitance.

3) Output Stage: Once the logic values on the inputs have been identified, they can be used to compute and drive out the correct reference signal on the gate output. To be able to achieve a full rail swing, it is necessary to have a complimentary pass gate structure. Figure 1 highlights the pass gate structure of the output stage of a simple inverter. To enable correct functionality we appropriately connect the correlator outputs to the reference signals in the pass gate

structure. The positive correlation outputs (V_1 in Figure 1) drive the NMOS gates of the pass gate structure while the negative correlation outputs (V_0 in Figure 1) drive the PMOS gates of the pass gate structure. To implement a buffer, the S_0 and S_1 connections in the pass gate structure (output stage) are reversed.

4) Complex Gates: In the same manner as described for an inverter, more complex gates can also be designed. Figure 2 illustrates the design of a sinusoidal supply based NAND2 gate. Both the inputs of the NAND2 gate are correlated with both reference signals to identify the logic values on the input signals. Again, a complimentary pass gate structure (shown in Figure 2) is used to drive out the correct reference signal.

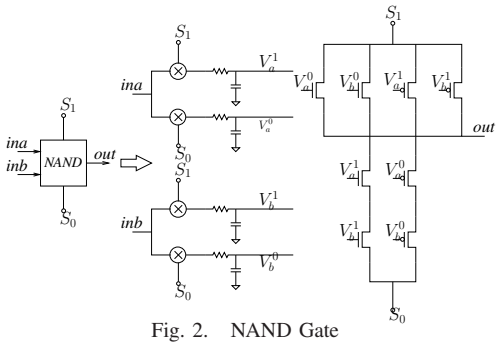

Fig. 2. NAND Gate

C. Interconnects

In the sinusoidal supply based logic style, information propagates around the circuit as high frequency AC signals, and these signals are correlated. Hence it is important to estimate the effect of interconnect delays on the signals. All realistic circuits are plagued with parasitic capacitances and inductances. The effect of these parasitics on a signal can be modeled as a transformation of these signals by the transfer function of the parasitics. Identical signals could reach a gate by traversing different paths (i.e. different parasitics). These signals could therefore have different phases and different magnitudes when they are correlated. We attempt to quantify the error induced by such a phase shift. If the input signal gets phase shifted by α and is correlated with the reference signals, its correlation reduces by a factor of $cos(\alpha)$, from Equation 6.

$$\frac{\overline{I}_\alpha}{\overline{I}_0} = \frac{cos(\alpha)\beta V_1^2}{cos(0)\beta V_1^2} = cos(\alpha)$$

As the phase difference increases, the correlation between signals from the same source decreases. To avoid this effect, the frequency of the reference sinusoidal signal should be chosen to be as low as possible. A fixed time shift of a signal presents itself as a smaller phase shift at lower frequencies, and hence we can tolerate more time shift without loosing correlation. But lower frequencies require a larger RC product in the low pass filter of the gate. A larger RC product leads to a slower gate, and also requires a larger area to implement R and C. This presents us with an optimization problem which we attempt to solve empirically in the next section.

IV. Experimental Results

In all our experiments we use a Predictive Technology Model (PTM) of 45nm metal gate, high-K dielectric, strained silicon model card [9]. All the sinusoid signals (unless otherwise stated) are 1 V peak to peak signals, with an average voltage of 0.5V. This choice of average voltage is driven by the fact that the resonant oscillator that produces the sinusoidal supplies produces sinusoids whose average value is $\frac{V_{DD}}{2}$. The choice of the amplitude is technology dependent but the technique of sinusoidal supply based circuit design is not restricted by the choice of peak to peak voltage.

We first quantify the power consumed in a circuit designed with sinusoidal supply based gates. The total power consumed is the power consumed in the gates as well as the power consumed in the standing wave oscillator [7].

A. Sinusoid Generator

We modified the sinusoid generator described in Section III-A to consume as little power as possible. Results from [7] indicate that the power consumed is a few milliwatts. However these power numbers are from experiments with very little capacitive load on the oscillator. However, large number of sinusoidal supply based gates are connected to the oscillator, they load the oscillator to a greater extent. As the load on the oscillator increases, the frequency of oscillation decreases and the power consumption increases. Figure 3 plots the variation in power and frequency of the oscillator as the load on the oscillator (in terms of the number of gates connected to the oscillator) increases. The nature of the load presented by the gates on the oscillator is capacitive, and it arises from the diffusion and gate terminals of the transistor of the gate. The frequency of oscillation varies with capacitive load C on the oscillator as $F = \frac{1}{\sqrt{LC}}$. On the other hand, the only source of power loss in the oscillator is the resistive loss. Power consumed due to resistive losses is I^2R, where I is the transient current whose value is determined by the characteristic impedance of the transmission line used in the oscillator.

$$I = \frac{V}{Z_0}$$

The characteristic impedance of a transmission line is given by

$$Z_0 = \sqrt{\frac{L}{C}}$$

, where L is the parasitic inductance in the transmission line and C is the parasitic capacitance. Thus the power consumed in the ring is

$$P = I^2R = \left(\frac{V}{Z_0}\right)^2 R$$

$$P = \frac{V^2C}{L}R = \frac{V^2CR}{L} \tag{8}$$

The power consumed in the oscillator is thus expected to grow linearly with the capacitive load on the oscillator. This is confirmed in Figure 3, which plots the power of the oscillator from HSPICE [10] simulations. In this simulation the NMOS devices of the inverter pair had a width of 40 μm. The ring was implemented in metal 8 and its length was 80 μm. This trend of increasing power with load continues until the negative resistance structure can no longer compensate for the losses in the oscillator ring.

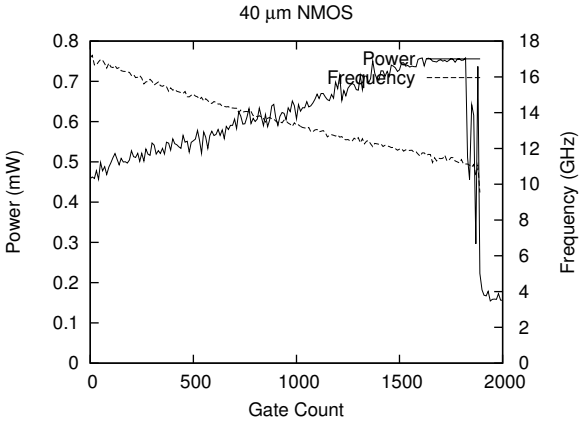

Fig. 3. Change in Frequency and Power with Load

B. Gate Optimization

As discussed in the Section III-B.2 the design of the low pass filter in a gate is an optimization problem. The optimal low pass filter design is dependent on the frequency of operation. To find the best frequency of operation we compare the performance of a simple gate (buffer) with a fanout of four at various frequencies. The output stage of a gate also acts as a low pass filter (the resistance of the output stage transistors followed by their diffusion capacitance and gate capacitance of the fanout gates forms the low pass filter).

The choice of frequency is process dependent. Choosing a low frequency increases output amplitude and reduces power consumption but also increases gate delay and the value of the filter's resistance and capacitance. The increase in switching power with increasing frequency and decrease in gate delay with increasing frequency can be observed in Figure 4,

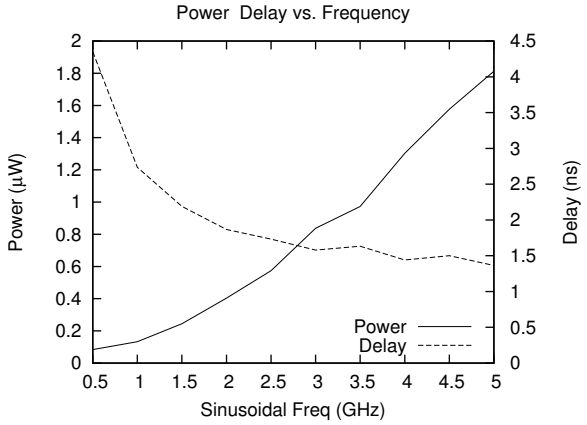

Fig. 4. Change in Power & Delay with Frequency

The choice of frequency is based on circuit performance requirements. For the rest of the discussion, we choose an operating frequency of 1 GHz. The next optimization explored is the choice of the R and C values in the low pass filter of a gate. As discussed in Section III-B.2 a large RC product increases the delay of computation, while a small RC product would make the voltage on the capacitor have a larger AC component. Mixing two 1 GHz sinusoids results in a DC component and a 2 GHz component. Thus the low pass filter has to be designed to suppress the 2 GHz frequency signals. Realizing large on-chip capacitance require a large amount of active area. Thus area is minimized by realizing a larger resistance. A large resistance can be implemented in resistive poly-silicon which offers a resistance of a few $M\Omega$ per square [11]. A 1 $M\Omega$ of resistance can therefore be practically realized on-chip with a small area footprint using poly-silicon. The gate capacitance of the minimum size device in the 45-nm technology node is of the order of 0.1 fF [11]. Thus an RC product of about 1 ns can be achieved with very low area.

To be able to suppress a signal of frequency 2 GHz we implement a low pass filter with a corner frequency of $\frac{2}{3}$ GHz. The filter has a resistor of 1 MΩ. The capacitance C is realized as the sum of gate capacitance presented by output stage and a dedicated capacitor in the form of gate capacitance, such that the RC product is 1.5 ns.

C. Gate Operation

We implemented and simulated (in HSPICE), simple gates based on the optimizations discussed. Simulation results of these gates are presented next. Figure 5 illustrates the operation of a simple buffer switching its output. The solid line represents the actual output of the buffer, while the dashed line is the reference signal that the buffer is expected to drive out. The output signal takes a non-sinusoid form when the output of the correlator is switching, due to a change in the input signal. The switching delay of the sinusoidal supply based inverter was about 2.55 ns.

We now compare the power of sinusoid based logic gates with traditional static CMOS gates. We present numbers from HSPICE simulations of a static CMOS inverter and a sinusoidal supply based inverter, both implemented in a 45nm PTM process. The static CMOS inverter is minimum sized, balanced and is made of devices with same threshold voltage as the devices in the output stage of the sinusoid gate. The static CMOS inverter is operated with 1 V supply. The power consumed over 1 ns (1 GHz clock) duration, during which the inverter switches once is 1.75 μW. The power consumed in a sinusoidal sypply based gate with operating frequency of 1 GHz for 1 computation is 500 nW, which is lower than the static CMOS gate. The leakage power of the static CMOS inverter at 1 V supply is 17 nW as opposed to 120nW for a sinusoidal supply based inverter. The high leakage power of sinusoidal supply based inverter is due to

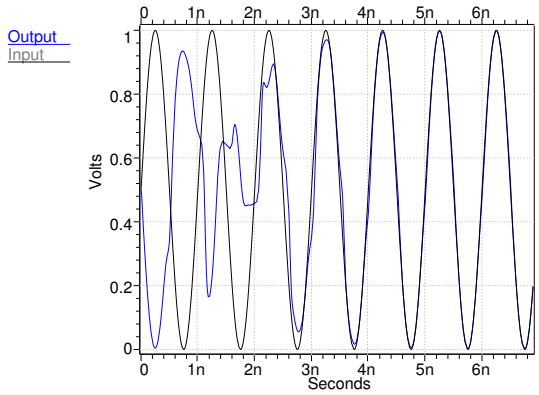

Fig. 5. Buffer Operation

the fact that the output stage switches from rail to rail even during steady state. Therefore the sinusoidal supply based gates have a lower dynamic power (power per computation) and high leakage power, and are thus more suitable for implementing high activity circuits. Al though the delay and the power-delay product (PDP) of the sinusoidal supply based gates are larger than their CMOS counterparts (for binary valued gates), we expect that using multivalued sinusoidal supply based gates would improve the PDP due to the possible reduction in logic depth.

V. Conclusions

Sinusoidal supply based logic could be a viable means to implement logic. They exhibit several advantages, being significantly immune to additive noise while elegantly implementing multi-valued logic. Since sinusoids from any two frequencies are orthogonal, multiple sinusoid signals can share the same electrical line and their logic values can still be recovered. Also, since additive noise does not affect the frequency of the signal, the signals could traverse a long distance without requiring to be buffered, thereby alleviating the problem of signal integrity. It should be noted that the two logic states used in this paper are not orthogonal but complimentary. Though we present results with complimentary signals the approach can be extended to orthogonal vector logic by employing multiple frequencies. We present design considerations in this paper to aid in designing sinusoidal supply based logic circuits. Our circuit level simulations indicate that sinusoidal supply based logic gates consume lower switching power than the static CMOS counterparts.

References

[1] L. B. Kish, "Thermal noise driven computing," *Applied Physics Letters*, vol. 89, no. 144104, 2006.

[2] L. B. Kish, "Noise-based logic: Binary, multi-valued, or fuzzy, with optional superposition of logic states," `http://arxiv.org/pdf/0808.3162`, 2008.

[3] F. Peper, "Noise-Driven Computation and Communication," *The Society of Instrument and Control Engineers (SICE), Japan, Catalog*, pp. 23–26, Nov. 2007.

[4] M. Ueda, M. Ueda, H. Takagi, M. Sato, T. Yanagida, I. Yamashita, and K. Setsune, "Biologically-inspired stochastic vector matching for noise-robust information processing," *Physica*, pp. 4475–4481, 2008.

[5] L. B. Kish, S. Khatri, and S. Sethuraman, "Noise-based logic hyperspace with the superposition of 2^N states in a single wire," *Physics Letters A*, pp. 1928–1934, 2009.

[6] F. O'Mahony, P. Yue, M. Horowitz, and S. Wong, "Design of a 10GHz clock distribution network using coupled standing-wave oscillators," in *DAC '03: Proceedings of the 40th conference on Design automation*, pp. 682–687, ACM, 2003.

[7] V. H. Cordero and S. P. Khatri, "Clock distribution scheme using coplanar transmission lines," in *Design Automation, and Test in Europe (DATE) Conference*, pp. 985–990, March 2008.

[8] V. Karkala, K. C. Bollapalli, R. Garg, and S. P. Khatri, "A PLL design based on a standing wave resonant oscillator," in *International Conference on Computer Design*, 2009.

[9] "Predictive Technology Model Homepage." `http://www.eas.asu.edu/~ptm/`.

[10] I. Meta-Software, "HSPICE user's manual," Campbell, CA.

[11] "The International Technology Roadmap for Semiconductors." `http://public.itrs.net/`, 2008.