

*Research Article*

# Difference-Equation-Based Digital Frequency Synthesizer

**Lu-Ting Ko,<sup>1</sup> Jwu-E. Chen,<sup>1</sup> Yaw-Shih Shieh,<sup>2</sup>  
Hsi-Chin Hsin,<sup>3</sup> and Tze-Yun Sung<sup>2</sup>**

<sup>1</sup> Department of Electrical Engineering, National Central University, Chungli 320-01, Taiwan

<sup>2</sup> Department of Electronics Engineering, Chung Hua University, Hsinchu 300-12, Taiwan

<sup>3</sup> Department of Computer Science and Information Engineering, National United University, Miaoli 360-03, Taiwan

Correspondence should be addressed to Tze-Yun Sung, bobsung@chu.edu.tw

Received 9 February 2012; Accepted 2 March 2012

Academic Editor: Ming Li

Copyright © 2012 Lu-Ting Ko et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a novel algorithm and architecture for digital frequency synthesis (DFS). It is based on a simple difference equation. Simulation results show that the proposed DFS algorithm is preferable to the conventional phase-locked-loop frequency synthesizer and the direct digital frequency synthesizer in terms of the spurious-free dynamic range (SFDR) and the peak-signal-to-noise ratio (PSNR). Specifically, the results of SFDR and PSNR are more than 186.91 dBc and 127.74 dB, respectively. Moreover, an efficient DFS architecture for VLSI implementation is also proposed, which has the advantage of saving hardware cost and power consumption.

## 1. Introduction

Many modern devices, for example, radio receivers, ADSL (Asymmetric Digital Subscriber Line), XDSL (X Digital Subscriber Line), 3G/4G mobile phones, walkie-talkies, CB radios, satellite receivers, and GPS systems, require frequency synthesizers with fine resolutions, fast channel switching, and large bandwidths. There are two types of frequency synthesizer available: phase-locked loop (PLL) and direct digital frequency synthesis (DDFS).

PLL is a control system, which generates an output signal with phase matched that of the input reference signal. Figure 1 shows the conventional PLL frequency synthesizer consisting of a phase detector, a charge pump, a lowpass filter, a voltage control oscillator, and a frequency divider [1–8]. The lower frequency signal,  $F_{div}$ , obtained by dividing the output signal via the frequency divider, is compared with the reference signal,  $F_{ref}$ , in the phase

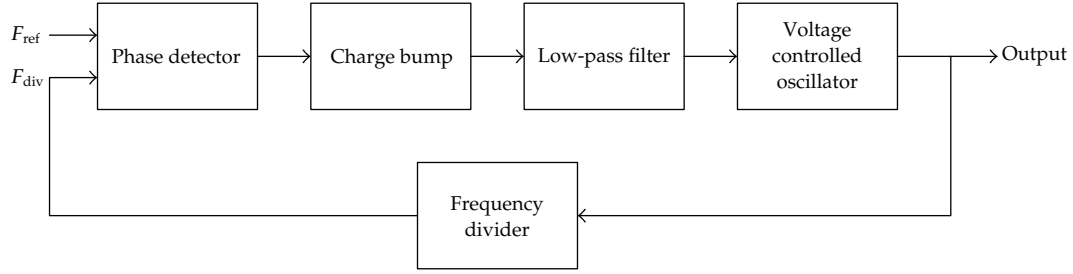


Figure 1: Block diagram of the conventional PLL system.

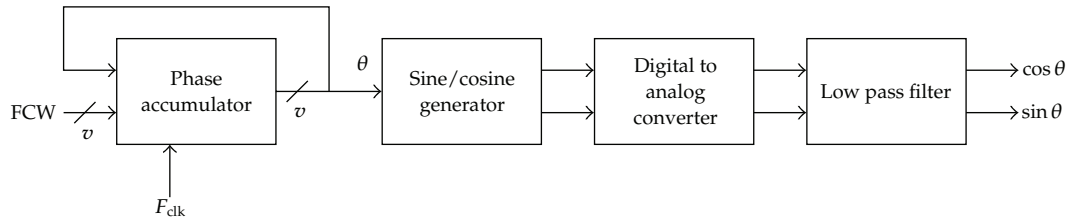


Figure 2: Block diagram of the conventional DDS architecture.

detector to generate an error signal, which is proportional to the phase difference. The charge bump converts the error signal pulse into analog current pulses, which are then integrated by using the lowpass filter, and drives the voltage-controlled oscillator to obtain the desired frequency.

The commonly used architecture of DDS [9] shown in Figure 2 consists of a phase accumulator, a sine/cosine generator, a digital-to-analog converter (DAC), and a lowpass filter (LPF). It takes two inputs: a reference clock and a  $v$ -bit frequency control word (FCW). In each clock cycle, the phase accumulator integrates FCW with periodical overflow to produce an angle in the range of  $[0, 2\pi)$ , the sine/cosine generator computes its sinusoidal value, which in practice is implemented digitally and, therefore, follows by DAC and LPF [10–23]. Various fractional-order ideal filters and fractional oscillators were proposed in [24–29].

Instead of using the conventional methods above, we propose a novel digital frequency synthesis (DFS) algorithm based on a simple difference equation. The rest of the paper is organized as follows. In Section 2, a novel DFS algorithm is proposed. In Section 3, the VLSI (very large-scale integration) digital frequency synthesizer is presented. In Section 4, the FPGA implementation and the performance evaluation are given. Conclusion can be found in Section 5.

## 2. The Proposed DFS Algorithm

The difference equation of DFS is as follows:

$$y[n-2] - 2a_1y[n-1] + a_2y[n] = x[n]. \quad (2.1)$$

Thus, we have the following characteristic equation:

$$z^{-2} - 2a_1z^{-1} + a_2 = 0. \quad (2.2)$$

The eigen-functions of (2.2) are represented as

$$z_{1,2} = re^{\pm j\theta}. \quad (2.3)$$

The ZIR (zero-input response) of DFS can be written as

$$\text{ZIR} = r^n (B_1 \cos(n\theta) + B_2 \sin(n\theta)), \quad (2.4)$$

where  $B_1$  and  $B_2$  are determined by initial conditions, and  $n = 0, 1, 2, \dots$

For DFS with sine wave generator, we have

$$r = 1, \quad B_1 = 0, \quad B_2 = 1, \quad \text{ZIR} = \sin(n\theta). \quad (2.5)$$

The eigen-functions of DFS are therefore as follows:

$$z_{1,2} = e^{\pm j\theta} = \cos \theta \pm j \sin \theta = c \pm jd. \quad (2.6)$$

Thus, the characteristic equation can be expressed as

$$z^{-2} - 2cz^{-1} + c^2 + d^2 = 0, \quad (2.7)$$

where  $c^2 + d^2 = 1$ .

Equation (2.7) could be rewritten as

$$z^{-2} - 2cz^{-1} + 1 = 0, \quad (2.8)$$

and the transfer function of DFS can be derived as

$$H(z) = \frac{1}{z^{-2} - 2cz^{-1} + 1}. \quad (2.9)$$

According to (2.9), the corresponding difference equation could be derived as

$$y[n] = x[n] - y[n-2] + 2c \cdot y[n-1], \quad (2.10)$$

where

$$\begin{aligned} y[0] &= B_1 = 0, \\ y[1] &= -B_1 \cos \theta + B_2 \sin \theta = d. \end{aligned} \quad (2.11)$$

As one can see, a rotation of angle  $\phi$  in the circular coordinate system can be obtained by performing a sequence of microrotations in an iterative manner. In particular, a vector can be successively rotated through the use of a sequence of predetermined step angles:  $\alpha(i) = \tan^{-1}(2^{-i})$ . This technique can be applied to generate many elementary functions, in which only simple adders and shifters are required. Thus, the well-known coordinate rotation digital computer (CORDIC) algorithm can be used for the DFS applications. The conventional CORDIC in the circular coordinate system is as follows [36–39]:

$$\begin{aligned} u(i+1) &= u(i) - \sigma(i)2^{-i}v(i), \\ v(i+1) &= v(i) + \sigma(i)2^{-i}u(i), \\ w(i+1) &= w(i) - \sigma(i)\alpha(i), \\ \alpha(i) &= \tan^{-1}2^{-i}, \end{aligned} \quad (2.12)$$

where  $\sigma(i) \in \{-1, +1\}$  denotes the direction of the  $i$ th microrotation,  $\sigma_i = \text{sign}(w(i))$  with  $w(i) \rightarrow 0$  in the vector rotation mode,  $\sigma_i = -\text{sign}(u(i)) \cdot \text{sign}(v(i))$  with  $v(i) \rightarrow 0$  in the angle accumulated mode, the corresponding scale factor  $k(i)$  is equal to  $\sqrt{1 + \sigma^2(i)2^{-2i}}$ , and  $i = 0, 1, \dots, l-1$ . The product of all scale factors after  $l$  microrotations is given by

$$K_1 = \prod_{i=0}^{n-1} k(i) = \prod_{i=0}^{n-1} \sqrt{1 + 2^{-2i}}. \quad (2.13)$$

In the vector rotation mode,  $\sin \phi$  and  $\cos \phi$  can be obtained, where the initial value  $(u(0), v(0)) = (1/K_1, 0)$ . In principle,  $u_{\text{out}}$  and  $v_{\text{out}}$  can be computed from the initial value  $(u_{\text{in}}, v_{\text{in}}) = (u(0), v(0))$  by using the following equation:

$$\begin{bmatrix} u_{\text{out}} \\ v_{\text{out}} \end{bmatrix} = K_1 \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} u_{\text{in}} \\ v_{\text{in}} \end{bmatrix}. \quad (2.14)$$

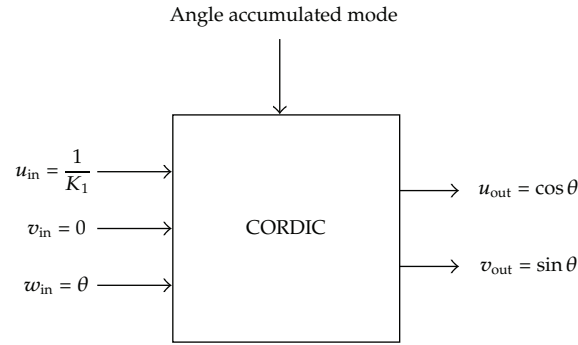
In order to evaluate the sinusoidal parameters:  $c$  and  $d$  for the proposed digital frequency synthesizer, the inputs of the CORDIC processor are  $u_{\text{in}} = 1/K_1$ ,  $v_{\text{in}} = 0$ , and  $w_{\text{in}} = \theta$  as shown in Figure 3.

### 3. Proposed Architecture for Digital Frequency Synthesizer

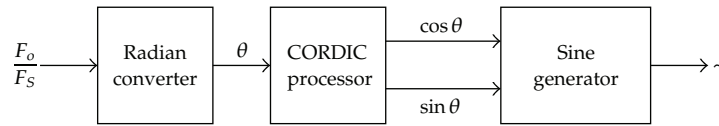
In this section, the architecture and the terminology associated with the proposed digital frequency synthesizer are presented. Our scheme is based on the proposed DFS algorithm combined with a CORDIC processor. It consists mainly of the radian converter, the CORDIC processor, and the sine generator as shown in Figure 4.

Figure 5 shows the radian converter. It is a constant multiplier, which converts the input signal into radians. Figure 6 shows the CORDIC processor, which evaluates the sinusoidal value and consists of three adders and two shifters.

Figure 7 shows the architecture of sine generator, which is the core of the proposed digital frequency synthesizer. It consists of one multiplier, one adder, and two latches only.



**Figure 3:** The CORDIC arithmetic for the proposed digital frequency synthesizer.



**Figure 4:** The proposed digital frequency synthesizer.

The key terminologies associated with the proposed digital frequency synthesizer are as follows.

### 3.1. Output Frequency

The output frequency of the proposed digital frequency synthesizer is determined by the coefficients  $d$  and  $c$ , since

$$\theta = \omega T_s = \tan^{-1}\left(\frac{d}{c}\right), \tag{3.1}$$

$$F_o = \frac{1}{2\pi} \cdot \tan^{-1}\left(\frac{d}{c}\right) \cdot F_s.$$

### 3.2. Frequency Resolution

For  $m$ -bit digital frequency synthesizer, the minimum change of the output frequency  $\Delta F_{o,\min}$  is expressed as

$$\Delta F_{o,\min} = \frac{1}{2\pi} \cdot \tan^{-1}\left(2^{-(m-1)}\right) \cdot F_s. \tag{3.2}$$

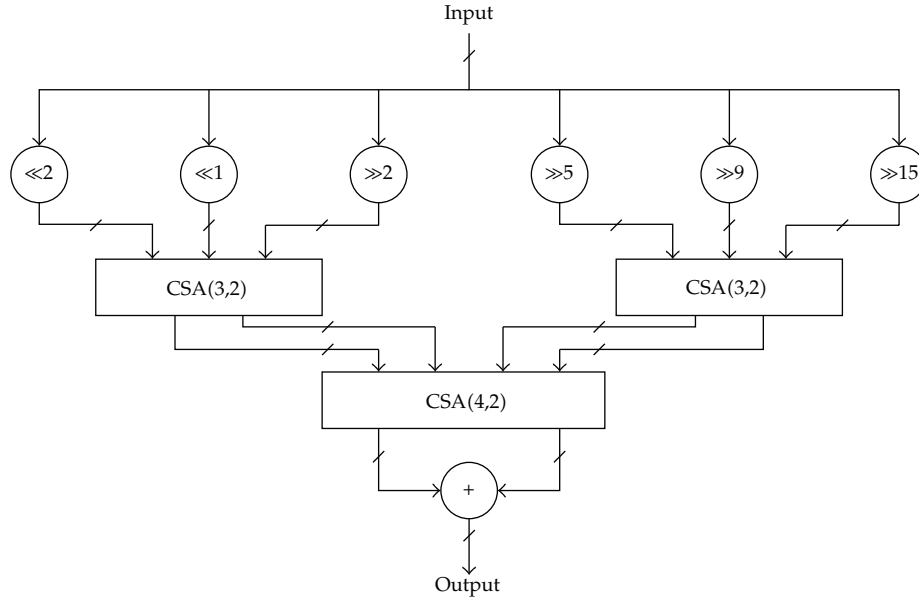


Figure 5: The radian converter.

### 3.3. Bandwidth

The bandwidth of digital frequency synthesizer is defined as the difference between the highest and lowest attainable output frequencies, which are expressed as follows:

$$F_{o,\max} = \frac{1}{2\pi} \cdot \tan^{-1}(1) \cdot F_s, \quad (3.3)$$

$$F_{o,\min} = \frac{1}{2\pi} \cdot \tan^{-1}\left(2^{-(m-1)}\right) \cdot F_s.$$

### 3.4. Peak Signal-to-Noise Ratio (PSNR)

A good direct digital frequency synthesizer should have an output signal with low noise, which can be evaluated by using the following signal-to-noise-ratio (PSNR) measured in dB:

$$\text{PSNR} = 20 \log\left(\frac{255}{\sqrt{\text{MSE}}}\right), \quad (3.4)$$

where MSE is the mean square error.

### 3.5. Spurious-Free Dynamic Range (SFDR)

The spurious-free dynamic range (SFDR) is defined as the ratio of the amplitude of the desired frequency component to that of the largest undesired frequency component at the output of a DDS. It is expressed in decibels (dBc) as follows:

$$\text{SFDR} = 20 \log\left(\frac{A_p}{A_s}\right) = 20 \log(A_p) - 20 \log(A_s), \quad (3.5)$$

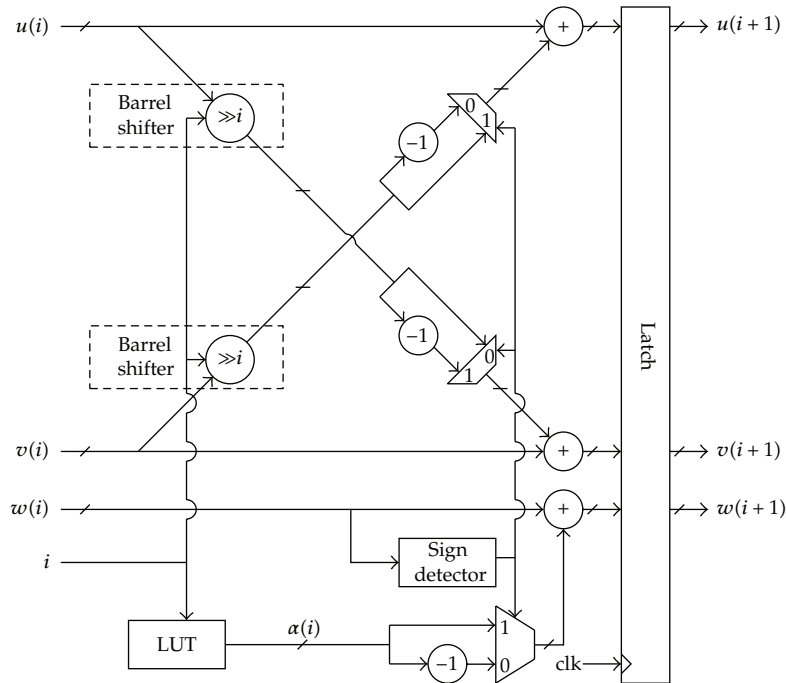


Figure 6: The CORDIC processor (LUT: Lookup table).

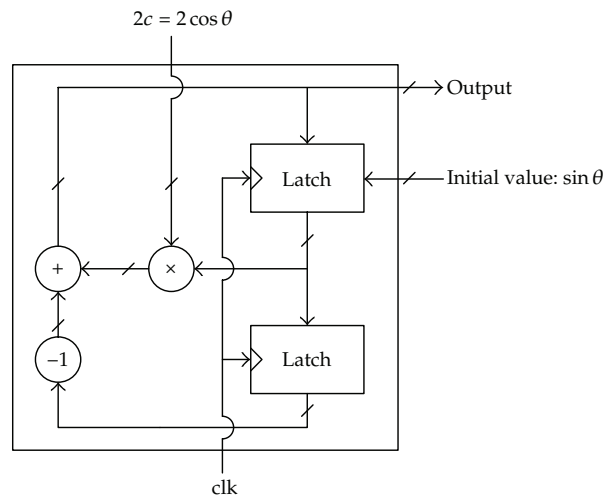


Figure 7: The sine generator.

where  $A_p$  is the amplitude of the desired frequency component,  $A_s$  is the amplitude of the largest undesired frequency component, and the higher the better.

#### 4. FPGA Implementation of Digital Frequency Synthesizer

In this section, the proposed high-performance architecture of digital frequency synthesizer is presented. Figure 8 depicts the system block diagram. The PSNR and SFDR of the proposed

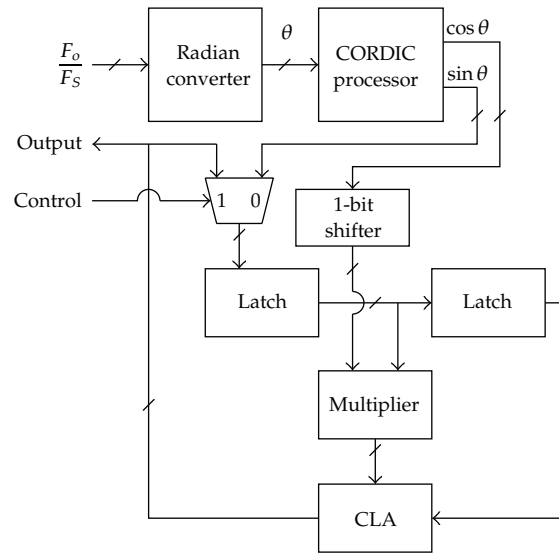


Figure 8: The proposed digital frequency synthesizer.

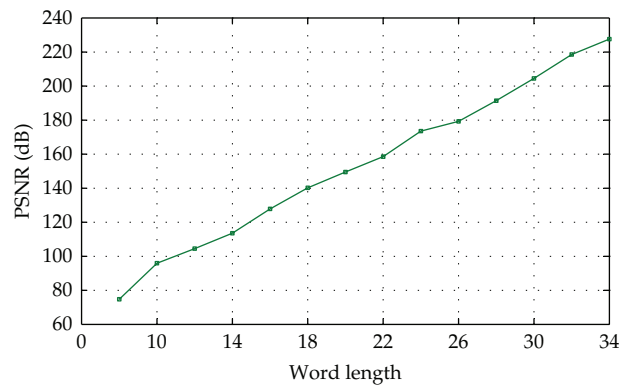


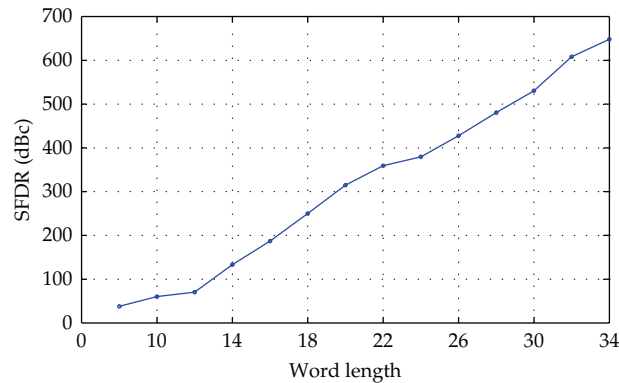
Figure 9: The PSNR of the proposed digital frequency synthesizer at various word lengths (100 MHz sampling rate and the maximum output frequency 12.5 MHz).

digital frequency synthesizer at various word lengths at 100 MHz sampling rate and the maximum output frequency 12.5 MHz are shown in Figures 9 and 10, respectively.

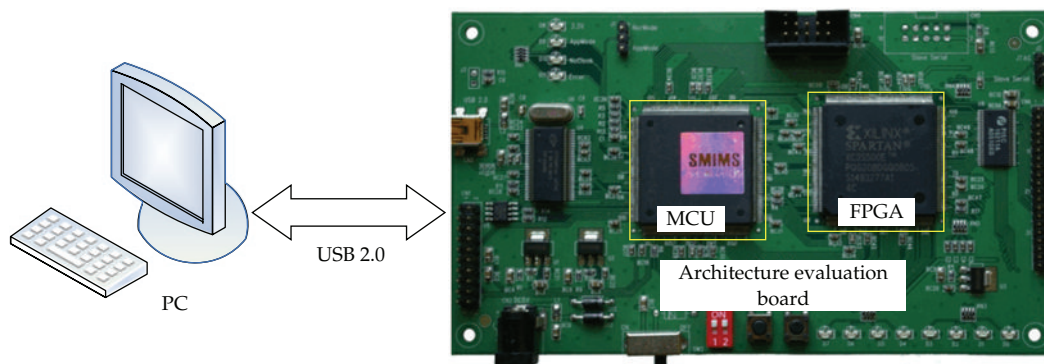
The platform for architecture development and verification has also been designed and implemented to evaluate the development cost. The proposed architecture of digital frequency synthesizer has been implemented on the field programmable gate array (FPGA) emulation board [40]. The FPGA has been integrated with the microcontroller (MCU) and I/O interface circuit (USB 2.0) to form the architecture development and verification platform.

Figure 11 depicts the block diagram and circuit board of the architecture development and evaluation platform. In which, the microcontroller reads data and commands from PC and writes the results back to PC via USB 2.0 bus; the FPGA implements the proposed





**Figure 10:** The SFDR of the proposed digital frequency synthesizer at various word lengths (100 MHz sampling rate and the maximum output frequency 12.5 MHz).



**Figure 11:** Block diagram and circuit board of the architecture development and verification platform.

architecture of digital frequency synthesizer. The hardware code in portable hardware description language runs on PC with the logic circuit simulator [41] and FPGA compiler [42]. It is noted that the throughput can be improved by using the proposed pipelined architecture while the computation accuracy is the same as that obtained by using the conventional architecture with the same word length. Thus, the proposed digital frequency synthesizer improves the power consumption and performance significantly. Moreover, all the control signals are internally generated onchip. The proposed digital frequency synthesizer provides a high-performance sinusoid waveform.

## 5. Conclusion

In this paper, we present a novel digital frequency synthesizer based on a simple difference equation with pipelined data path. Circuit emulation shows that the proposed high-performance architecture has the advantages of high precision, high data rate, and simple hardware. For 16-bit digital frequency synthesizer, the PSNR and SFDR obtained by using the proposed architecture at the maximum output frequency are 127.74 dB and 186.91 dBc, respectively. As shown in Table 1, the proposed digital frequency synthesizer is superior to the previous works in terms of SFDR, PSNR, and hardware [18, 30–35]. The proposed digital

**Table 1:** Comparisons between the proposed DFS and other related works.

Authors	Items					
	Output resolution (bits)	SFDR (dBc)	PSNR (dB)	ROM (words)	Adders	Multipliers
Strollo et al. [18]	13	90.2	—	1344	8	0
Song and Kim [30]	16	100	—	270	6	6
Langlois and Al-Khalili [31]	14	96.2	—	1152	8	0
De Caro et al. [32]	12	80	—	0	2	2
De Caro and Strollo [33]	12	83.6	—	896	2	2
Yang et al. [34]	12	80	—	2176	6	0
Curticapean and Niittylahti [35]	14	85	—	832	2	2
Ko et al. (This work)	14	133	113	14	8	1
	16	187	128	16	8	1

frequency synthesizer designed by portable hardware description language is a reusable IP, which can be implemented in various VLSI processes with trade-offs of performance, area and power consumption.

## Acknowledgment

The National Science Council of Taiwan, under Grants NSC100-2628-E-239-002-MY2, supported this work.

## References

- [1] W. Chen, V. Pouget, H. J. Barnaby et al., "Investigation of single-event transients in voltage-controlled oscillators," *IEEE Transactions on Nuclear Science*, vol. 50, no. 6, pp. 2081–2087, 2003.
- [2] F. M. Gardner, *Phase-Lock Techniques*, Wiley, New York, NY, USA, 3rd edition, 2005.
- [3] J. Y. Chang, C. W. Fan, C. F. Liang, and S. I. Liu, "A single-PLL UWB frequency synthesizer using multiphase coupled ring oscillator and current-reused multiplier," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 56, no. 2, pp. 107–111, 2009.
- [4] H. H. Chung, W. Chen, B. Bakaloglu, H. J. Barnaby, B. Vermeire, and S. Kiaei, "Analysis of single events effects on monolithic PLL frequency synthesizers," *IEEE Transactions on Nuclear Science*, vol. 53, no. 6, pp. 3539–3543, 2006.
- [5] T. Wu, P. K. Hanumolu, K. Mayaram, and U. K. Moon, "Method for a constant loop bandwidth in LC-VCO PLL frequency synthesizers," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 2, pp. 427–435, 2009.
- [6] R. B. Staszewski, "State-of-the-art and future directions of high-performance all-digital frequency synthesis in nanometer CMOS," *IEEE Transactions on Circuits and Systems. I. Regular Papers*, vol. 58, no. 7, pp. 1497–1510, 2011.
- [7] M. Zanuso, P. Madoglio, S. Levantino, C. Samori, and A. L. Lacaita, "Time-to-digital converter for frequency synthesis based on a digital bang-bang DLL," *IEEE Transactions on Circuits and Systems. I. Regular Papers*, vol. 57, no. 3, pp. 548–555, 2010.
- [8] R. B. Staszewski, S. Vemulapalli, P. Vallur, J. Wallberg, and P. T. Balsara, "1.3 V 20 ps time-to-digital converter for frequency synthesis in 90-nm CMOS," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 3, pp. 220–224, 2006.
- [9] J. Tierney, C. Rader, and B. Gold, "A digital frequency synthesizer," *IEEE Transactions on Audio and Electroacoustics*, vol. 19, no. 1, pp. 48–57, 1971.

- [10] A. Bonfanti, D. De Caro, A. D. Grasso, S. Pennisi, C. Samori, and A. G. M. Strollo, "A 2.5-GHz DDFS-PLL with 1.8-MHz bandwidth in 0.35- $\mu$ m CMOS," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 6, pp. 1403–1413, 2008.
- [11] D. De Caro, E. Napoli, and A. G. M. Strollo, "Direct digital frequency synthesizers with polynomial hyperfolding technique," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 51, no. 7, pp. 337–344, 2004.
- [12] Y. H. Chen and Y. A. Chau, "A direct digital frequency synthesizer based on a new form of polynomial approximations," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 436–440, 2010.
- [13] C. Y. Kang and E. E. Swartzlander, "Digit-pipelined direct digital frequency synthesis based on differential CORDIC," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 53, no. 5, pp. 1035–1044, 2006.
- [14] D. De Caro, N. Petra, and A. G. M. Strollo, "High-performance special function unit for programmable 3-D graphics processors," *IEEE Transactions on Circuits and Systems. I. Regular Papers*, vol. 56, no. 9, pp. 1968–1978, 2009.
- [15] D. Yang, F. F. Dai, W. Ni, S. Yin, and R. C. Jaeger, "Delta-sigma modulation for direct digital frequency synthesis," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 17, no. 6, pp. 793–802, 2009.
- [16] T. Rapinoja, K. Stadius, L. Xu et al., "A digital frequency synthesizer for cognitive radio spectrum sensing applications," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 5, pp. 1339–1348, 2010.
- [17] A. McEwan and S. Collins, "Direct digital-frequency synthesis by analog interpolation," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 11, pp. 1294–1298, 2006.
- [18] A. G. M. Strollo, D. De Caro, and N. Petra, "A 630 MHz, 76 mW direct digital frequency synthesizer using enhanced ROM compression technique," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 2, pp. 350–360, 2007.
- [19] Y. D. Wu, C. M. Lai, C. C. Lee, and P. C. Huang, "A quantization error minimization method using DDS-DAC for wideband fractional-N frequency synthesizer," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 11, pp. 2283–2291, 2010.
- [20] S. E. Turner, R. T. Chan, and J. T. Feng, "ROM-based direct digital synthesizer at 24 GHz clock frequency in InP DHBT technology," *IEEE Microwave and Wireless Components Letters*, vol. 18, no. 8, pp. 566–568, 2008.
- [21] T. Y. Sung, L. T. Ko, and H. C. Hsin, "Low-power and high-SFDR direct digital frequency synthesizer based on hybrid CORDIC algorithm," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '09)*, pp. 249–252, May 2009.
- [22] T. Y. Sung, H. C. Hsin, and L. T. Ko, "High-SFDR and multiplierless direct digital frequency synthesizer," *WSEAS Transactions on Circuits and Systems*, vol. 8, no. 6, pp. 455–464, 2009.
- [23] Y. J. Cao, Y. Wang, and T. Y. Sung, "A ROM-Less direct digital frequency synthesizer based on a scaling-free CORDIC algorithm," in *Proceedings of the 6th International Forum on Strategic Technology*, pp. 1186–1189, Harbin, China, 2011.
- [24] M. Li, "Approximating ideal filters by systems of fractional order," *Computational and Mathematical Methods in Medicine*, vol. 2012, Article ID 365054, 6 pages, 2012.
- [25] M. Li, S. C. Lim, and S. Chen, "Exact solution of impulse response to a class of fractional oscillators and its stability," *Mathematical Problems in Engineering*, vol. 2011, Article ID 657839, 2011.
- [26] M. Li, C. Cattani, and S. Y. Chen, "Viewing sea level by a one-dimensional random function with long memory," *Mathematical Problems in Engineering*, vol. 2011, Article ID 654284, 13 pages, 2011.
- [27] E. G. Bakhoun and C. Toma, "Specific mathematical aspects of dynamics generated by coherence functions," *Mathematical Problems in Engineering*, vol. 2011, Article ID 436198, 10 pages, 2011.
- [28] S. Y. Chen, J. Zhang, H. Zhang, N. M. Kwok, and Y. F. Li, "Intelligent lighting control for vision-based robotic manipulation," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 8, pp. 3254–3263, 2012.
- [29] S. Y. Chen, H. Tong, and C. Cattani, "Markov models for image labeling," *Mathematical Problems in Engineering*, vol. 2012, Article ID 814356, 18 pages, 2012.
- [30] Y. Song and B. Kim, "Quadrature direct digital frequency synthesizers using interpolation-based angle rotation," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 7, pp. 701–710, 2004.
- [31] J. M. P. Langlois and D. Al-Khalili, "Novel approach to the design of direct digital frequency synthesizers based on linear interpolation," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 50, no. 9, pp. 567–578, 2003.

- [32] D. De Caro, E. Napoli, and A. G. M. Strollo, "Direct digital frequency synthesizers with polynomial hyperfolding technique," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 51, no. 7, pp. 337–344, 2004.
- [33] D. De Caro and A. G. M. Strollo, "High-performance direct digital frequency synthesizers using piecewise-polynomial approximation," *IEEE Transactions on Circuits and Systems. I. Regular Papers*, vol. 52, no. 2, pp. 324–337, 2005.
- [34] B. D. Yang, J. H. Choi, S. H. Han, L. S. Kim, and H. K. Yu, "An 800-MHz low-power direct digital frequency synthesizer with an on-chip D/A converter," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 5, pp. 761–774, 2004.
- [35] F. Curticapean and J. Niittylahti, "A hardware efficient direct digital frequency synthesizer," in *Proceedings of the IEEE International Conference Electronics, Circuits and Systems (ICECS '01)*, pp. 51–54, Malta, 2001.
- [36] J. Volder, "The CORDIC trigonometric computing technique," *IRE Transactions on Electronic Computers*, vol. 8, no. 3, pp. 330–334, 1959.
- [37] J.S. Walther, "A unified algorithm for elementary functions," in *Proceedings of the Spring Joint Computer Conference (AFIPS '71)*, pp. 379–385, 1971.
- [38] T. Y. Sung and H. C. Hsin, "Design and simulation of reusable IP CORDIC core for special-purpose processors," *IET Computers and Digital Techniques*, vol. 1, no. 5, pp. 581–589, 2007.
- [39] Y. H. Hu, "CORDIC-based VLSI architectures for digital signal processing," *IEEE Signal Processing Magazine*, vol. 9, no. 3, pp. 16–35, 1992.
- [40] SMIMS Technology Corp., <http://www.smims.com>.
- [41] S. Palnitkar, *Verilog HDL-A Guide to Digital Design and Synthesis*, Prentice-Hall, 2nd edition, 2003.
- [42] Xilinx FPGA products, 2011, <http://www.xilinx.com/products/>.





# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

