ROM-less DDFS Using Non-equal Division Parabolic Polynomial Interpolation Method

Chia-Hao Hsu, Yun-Chi Chen, and Chua-Chin Wang†, *Senior Member, IEEE*

Department of Electrical Engineering
National Sun Yat-Sen University
Kaohsiung, Taiwan 80424
Email: ccwang@ee.nsysu.edu.tw

***Abstract*— A direct digital frequency synthesizer (DDFS) based on a non-equal division parabolic polynomial interpolation method is proposed in this paper. To attain high spurious free dynamic range (SDRF) and reduce area cost, a parabolic polynomial interpolation method is adopted in the proposed design to replace conventional ROM-based phase-to-sine mapper methods. Particularly, the left 1/4 of the phase range is approximated using a low-curvature parabolic curve. The proposed design is manufactured using a standard 0.18 μm CMOS technology. The maximum output frequency is 50 MHz, the core area is 1.4528 mm^2, and the spurious free dynamic range (SFDR) is 68.67 dBc. The proposed DDFS outperforms prior works' SFDR and energy efficiency.**

***Keywords*—Direct digital frequency synthesizer (DDFS), parabolic polynomial interpolation, spurious free dynamic range (SFDR).**

I. Introduction

Frequency synthesizer is a well-known technique to generate signals with a selective frequency, which is an important component for many communication systems, e.g., digital radios, mobile telephones, GPSs (Global Positioning Systems), etc. Traditionally, a signal with a selective frequency is mostly generated by phase-locked loop (PLL) [1]- [2] circuits. However, two major drawbacks of the the PLL-based frequency generators are slow frequency switching speed and poor spectral purity [3]. The deficiencies of the PLL-based frequency generators are then inadequately used in the modern wireless communication systems to meet fast frequency switching demand. By contrast, the direct digital frequency synthesizer (DDFS) has been considered as a better alternative than the PLL-based frequency generators, because it is proved to provide fast frequency switching and excellent spectral purity.

The DDFS technique was first proposed in 1971's [4], which utilized digital data processing technique to generate a tunable frequency signal driven by a precise clock signal. Fig. 1 shows the architecture of the conventional DDFS. The phase accumulator is used to generate the digital phase as well as the frequency. The samples of sine wave amplitude are stored in the ROM-based look-up table, which are possibly derived from MATLAB simulation results. The samples are converted into digital sine wave signals by the amplitude complementor. Finally, the digital-to-analog converter (DAC) converts the digital sine wave signals into an analog sine wave signal. Unfortunately, the conventional DDFS has a major intrinsic difficulty: it demands very large ROM as the storage of the sine wave amplitude samples. Therefore, the conventional DDFS will suffer from three inherent drawbacks of the ROM, which are large power consumption, large chip area, and slow operating speed. Although the ROM size can be significantly reduced by truncating the output bit-wide of the phase accumulator, the unwanted spurious noise caused by the truncation will then degrade the spectral purity performance of the generated sine wave.

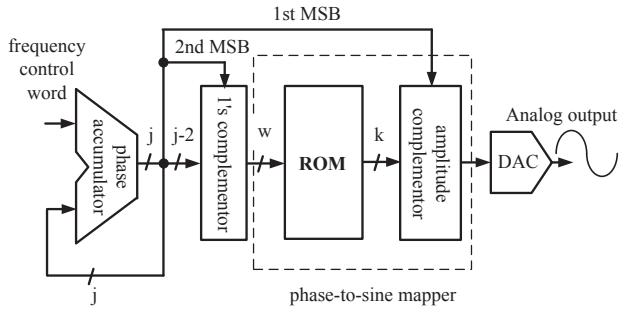

Fig. 1. Conventional DDFS structure

Lately, ROM-less DDFS [5]- [9] researches have been proposed to attain hardware reduction advantage, which utilized different high-order (probably more than three order) polynomial algorithms instead of the ROM-based look-up tables to realize the phase-to-sine mappers. Due to the intrinsic massive computation complexity, any algorithm based on high-order polynomials will be hard to meet the high speed requirement of modern wireless communication applications. Moreover, according to the reports of prior works [12], any polynomial whose order is larger than three may be inefficient to improve the spurious free dynamic range (SFDR) of the sine wave. This paper proposes a novel DDFS based on a 2nd-order non-equal

†Prof. C.-C. Wang is with the Department of Electrical Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan 80424. (e-mail: ccwang@ee.nsysu.edu.tw). He is the contact author.

division parabolic polynomial interpolation method to resolve all of the mentioned difficulties. Compared with several prior works, the proposed DDFS shows a satisfactory speed and an exceptional spectral purity.

II. The proposed DDFS Architecture

A. Parabolic Polynomial Interpolation

To implement the phase-to-sine mapper, the parabolic polynomial could be the most convenient choice to provide superior SFDR. A quadrant of sinusoid is partitioned into M segments, where every segment is approximated by the 2nd-order parabolic polynomial, as shown in Eqn. (1).

$$y(x) = a_i x^2 + c_i, \quad i = 1 \sim M. \tag{1}$$

where the parameters, a_i and c_i, can be derived by the least square method. However, large M will result in divergence when performing the least square method, though the SFDR of this DDFS will be enhanced as the increase of M. The parabolic polynomial method has difficulty in fitting the curvature of the sinusoid. The curvature at x, $k(x)$, of a given curve $y = f(x)$ can be given by

$$k(x) = \frac{y''}{(1 + y'^2)^{3/2}} \tag{2}$$

where the 1st-order derivative, y', is equal to $2a_i x$ and the 2nd-derivative, y'', is equal to $2a_i$. Obviously, y' is proportional to y''. Thus, the fitting procedure is difficult to determine both the slope and the curvature of the curve if only one parameter a_i is allowed.

To resolve the problem of Eqn. (2), this paper adopts another parabolic polynomial method. We propose to add a coefficient, x_i, in Eqn. (2). Then, Eqn. (2) is rewritten as Eqn. (3) to enhance the adjustability of the parabolic polynomial.

$$y(x) = a_i(x + x_i)^2 + c_i, \quad i = 1 \sim M \tag{3}$$

Therefore, the new 1st-order derivative formula is given by $y' = 2a_i(x + x_i)$. Regarding the computation complexity, Eqn. (3) needs only one more addition compared with Eqn. (1).

The coefficient, x_i, is derived by the variation between an ideal sine wave and the generated sine wave, which is used to adjust the characteristics of the curve in each segment to carry out the phase-to-sine mapping function. Notably, x_i, is given an initial value, "0", at the start state. Then, a_i and c_i are, respectively, derived by the least square method. Therefore, Eqn. (3) can be re-formulated into a matrix expression as follows.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix} \begin{bmatrix} a_i \\ c_i \end{bmatrix} \tag{4}$$

The transpose of the first matrix at the righthand side of Eqn. (4) is used to be multiplied at both sides of Eqn. (4) to attain the following equation.

$$\begin{aligned} &\begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} y_1 \\ y_2 \\ y_2 \\ \vdots \\ y_M \end{bmatrix} \\ &= \begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix} \begin{bmatrix} a_i \\ c_i \end{bmatrix} \end{aligned} \tag{5}$$

Eqn. (5) is then reorganized as follows.

$$\begin{aligned} \begin{bmatrix} a_i \\ c_i \end{bmatrix} = &\left(\begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix} \right)^{-1} \\ &\begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_M \end{bmatrix} \end{aligned} \tag{6}$$

Thus, the coefficients, a_i and c_i, can be derived based on Eqn. (6). Furthermore, the approximation value of the sine wave amplitude can be easily deduced according to Eqn. (4) and Eqn. (6), which is expressed as Eqn. (7).

$$\begin{bmatrix} y'_1 \\ y'_2 \\ y'_3 \\ \vdots \\ y'_M \end{bmatrix} = \begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix} \left(\begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix}^{\mathrm{T}} \right.$$
$$\left. \begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix} \right)^{-1} \begin{bmatrix} (x+x_1)^2 & 1 \\ (x+x_2)^2 & 1 \\ (x+x_3)^2 & 1 \\ \vdots & \vdots \\ (x+x_M)^2 & 1 \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_M \end{bmatrix} \quad (7)$$

Therefore, the least square error, x_i, is found according to Eqn. (7). Finally, we then back-substitute x_i into Eqn. (6) such that a_i and c_i can be derived, respectively.

In this study, a quadrant of sinusoid is divided into 16 segments ($M = 16$). To attain a better spectral purity, we analyze two different segment division methods to choose a better one so as to decide the range of each segment. As shown in Fig. 2(a), each segment is equally distributed over the entire phase range, which is called equal-segment division method. By contrast, Fig. 2(b) shows the proposed non-equal division method.

Fig. 3 depicts the absolute errors of equal-segment division method and non-equal division method, which are, respectively, derived by the difference between an ideal sine wave and the generated sine waves. Notably, the left 1/4 quadrant of the sinusoid ($\sin\theta$, $0 \le \theta < \pi/8$) is approximate by a low-curvature parabolic curve. By contrast, the right 3/4 quadrant of the sinusoid ($\sin\theta$, $\pi/8 \le \theta < \pi/2$) is approximated using a high-curvature parabolic curve with intensive samples to reduce the error. In short, the 1st segment, S1, contains 1/4 quadrant of the sinusoid at left side, while the remaining 3/4 quadrant of the sinusoid is equally divided into the other 15 segments. Therefore, the proposed non-equal division method will synthesize a better sine wave without paying too much hardware overhead.

Fig. 4 shows the schematic of the proposed DDFS, which utilizes a 32-bit frequency control word (FCW) to attain a wide frequency tuning range. The output of the squarer through the selectors is shifted by the shift registers, a_{i-j}, where the symbol a_{i-j} represents the j-th bit of the sequence for the i-th segment. Then, the outputs of the shift registers are selected by multiplexers (MUXs), respectively. The summation of the total 16 multiplexer's outputs is realized by an adder.

III. Implementation and Simulation

The proposed DDFS is realized on silicon using TSMC (Taiwan Semiconductor Manufacturing Company) standard 0.35 μm CMOS technology. All of the PVT corner simulations ([0^{o}C, +25^{o}C, +75^{o}C, +100^{o}C], (VDD, (1$\pm$10%)VDD), and SS, FS, TT, SF, FF models), have been carried out to justify its robustness. Fig. 5 illustrates the chip layout including I/O

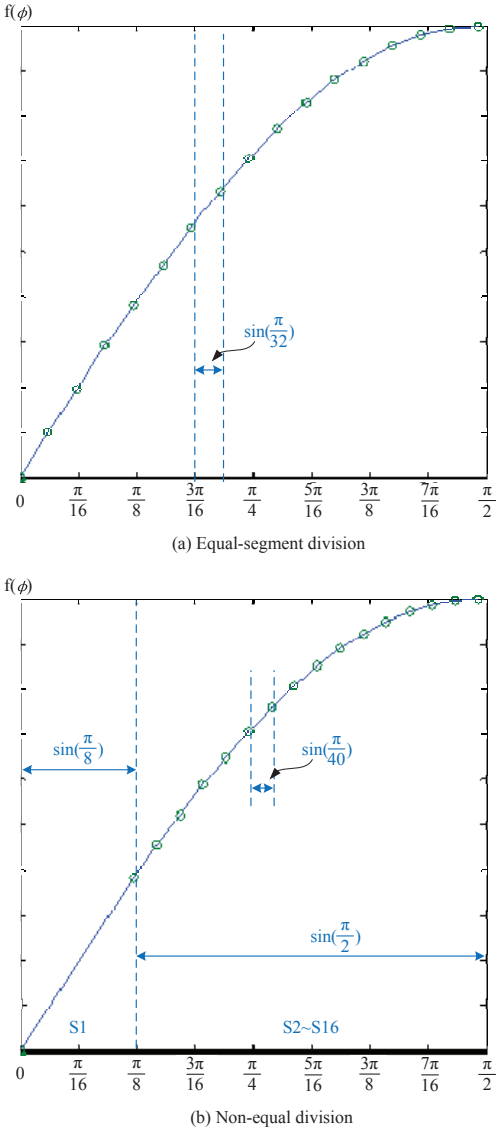

Fig. 2. Segment division methods

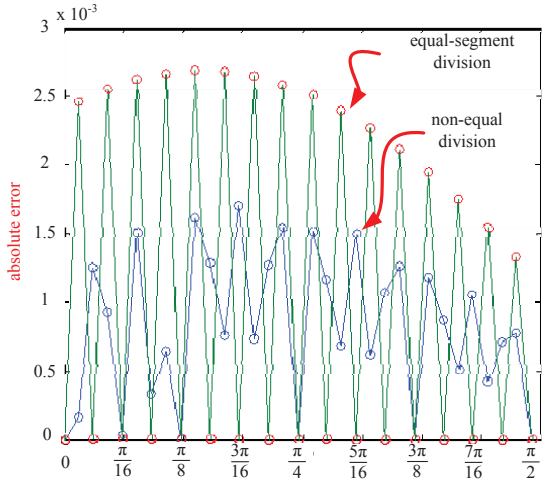

Fig. 3. Absolute error of two different segment division methods

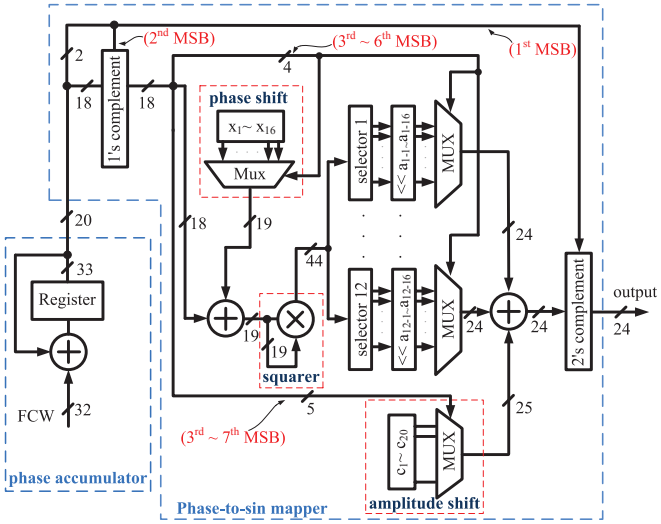

Fig. 4. Schematic of the proposed DDFS

PADs of the proposed DDFS design. The comparison between the proposed DDFS and prior DDFS designs is tabulated in Table I. The proposed DDFS attains the largest SFDR (68.67 dBc) and the best energy efficiency. The power consumption is 15 mW at a 50 MHz clock.

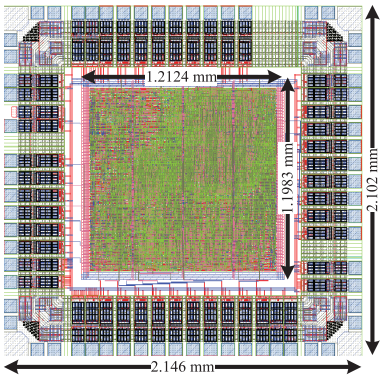

Fig. 5. Layout view of the proposed DDFS

IV. Conclusion

This paper proposes a ROM-less DDFS using a 16-segment non-equal division parabolic polynomial to realize the phase-to-sine mapping function. The proposed DDFS demonstrates a significant improvement in SFDR compared to prior works. Besides, the proposed DDFS is carried out as a multiplier-less design. The proposed DDFS attains a SFDR of 68.67 dBc, a 0.31 mW/MHz energy efficiency and the maximum clock rate is 50 MHz.

Acknowledgement

This investigation is partially supported by National Science Council under grant NSC 99-2221-E-110-081-MY3 and NSC 99-2923-E-110-0 02-MY2. Besides, this research is supported by the Southern Taiwan Science Park Administration (STSPA), Taiwan, R.O.C. under contract no. EZ-10-09-44-98. It is also partially supported by Ministry of Economic Affairs, Taiwan, under grant 99-EC-17-A-01-S1-104 and 99-EC-17-A-19-S1-133. The authors would like to express their deepest gratefulness to Chip Implementation Center of National Applied Research Laboratories, Taiwan, for their thoughtful chip fabrication service.

TABLE I

Comparison of DDFS designs

	[10]	[11]	[12]	ours
Process (μm)	0.35	*	0.13	0.35
Type	sine	quad	sine	sine
SFDR (dBc)	35	50.89	64.72	68.67
Phase accumulator bits	8	15	20	20
Output resolution (bits)	8	14	24	24
Power Efficiency (mW/MHz)	410000	1044.35	0.35	0.31
Max. Clock (MHz)	2000	11.25	161	50
Area (core) mm^2	N/A	N/A	0.33	1.4528
Area (with pad) mm^2	3.99	N/A	2.015	4.5109
Year	2005	2008	2009	2011

*The DDFS design was implemented by FPGA.

References

[1] Y. Sun, X. Yu, W. Rhee, D. Wang, and Z. Wang, "A fast settling dual-path fractional-N PLL with hybrid-mode dynamic bandwidth control," *IEE Microwave and Wireless Components Letters*, vol. 20, no. 8, pp. 462-464, Aug. 2010.

[2] W.-Y. Shin, M. Kim, G.-M. Hong, and S. Kim, "A fast-acquisition PLL using split half-duty sampled feedforward loop filter," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1856-1859, Aug. 2010.

[3] M. Kesoulis, D. Soudris, C. Koukourlis, and A. Thanailakis, "Systematic methodology for designing low power direct digital frequency synthesizers," *IET Circuits, Devices & Systems*, vol. 1, no. 4, pp. 293-304, Aug. 2007.

[4] J. Tierney, C. Rader, and B. Gold, "A digital frequency synthesizer," *IEEE Trans. on Audio and Electroacoustics*, vol. 19, no. 1, pp. 48-57, Mar. 1971.

[5] A. Ashrafi and R. Adhami, "A direct digital frequency synthesizer utilizing quasi-linear interpolation method," in Proc. of *37th IEEE Southeastern Symposium on System Theory*, pp. 144-148, Mar. 2005.

[6] A. Ashrafi and R. Adhami, "An optimized direct digital frequency synthesizer based on even fourth order polynomial interpolation," in Proc. of *38th IEEE Southeastern Symposium on System Theory*, pp. 109-113, Mar. 2006.

[7] H. Jafari, A. Ayatollahi, and S. Mirzakuchaki, "A low power, high SFDR, ROM-less direct digital frequency synthesize," in Proc. of *2005 IEEE Conf. on Electron Devices and Solid-State Circuits*, pp. 50-54, Dec. 2005.

[8] D. De Caro and A. G. M. Strollo, "High-performance direct digital frequency synthesizers in 0.25 μm CMOS using dual-slope approximation," *IEEE J. of Solid-State Circuits*, vol. 40, no. 11, pp. 2220-2227, Nov. 2005.

[9] Y. Song and B. Kim, "A 250 MHz direct digital frequency synthesizer with delta-sigma noise shaping," in Proc. of *2003 IEEE Int. Solid-State Circuits Conf. (ISSCC 2003)* vol. 1, pp. 472-509, Jan. 2003.

[10] X. Yu, F. F. Dai, Y. Shi, and R. Zhu, "2 GHz 8-bit CMOS ROM-less direct digital frequency synthesizer," in Proc. of *IEEE Inter. Symp. on Circuits and Systems, 2005. (ISCAS 2005)*, vol. 5, pp. 4397-4400, May 2005.

[11] S. S. Jeng, H. C. Lin, and C. Y. Wu, "DDFS design using the equi-section division method for SDR transceiver," in Proc. of *IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1-5, Sept. 2008.

[12] C.-C. Wang, C.-H. Hsu, C.-C. Lee, and J.-M. Huang, "A ROM-less DDFS based on a parabolic polynomial interpolation method with an offset," *Journal of Signal Processing Systems* vol. 61, pp.1-9, May 2010.