

A DIRECT DIGITAL FREQUENCY SYNTHESIZER BASED ON AUTOMATIC NONUNIFORM PIECEWISE FUNCTION GENERATION

Jochen Rust and Steffen Paul

Institute of Electrodynamics and Microelectronics (ITEM)
University of Bremen, Bremen, Germany
+49(0)421/218-62538 {rust, steffen.paul}@me.uni-bremen.de

ABSTRACT

As nowadays Direct Digital Frequency Synthesizers (DDFS) are used in a vast area of applications, the demand for simple and efficient hardware design and implementation methods is a highly important aspect. In this paper a new approach is introduced considering Automatic Nonuniform Piecewise linear function Approximation (ANPA). Automatic function generation is performed that enables quick HDL design by parameter specification in advance. For evaluation, several different configurations are simulated regarding approximation accuracy and complexity. In addition, logical and physical IC synthesis is performed for selected designs and their results are compared with actual references with respect to the common hardware constraints power, area and time.

Index Terms— Direct Digital Frequency Synthesizer, Nonuniform Piecewise Function Approximation

1. INTRODUCTION

The design of efficient Frequency Synthesizers (FS) in the scope of microelectronics has become more and more important in recent years, due to a large number of operation areas [1, 2, 3]. Basically, FS are used for accurate sine function calculation, which is used for e.g. synchronization and signal modulation in mobile communication [2], implantable circuits in medical technology [1] or in military spaceborne applications [3]. Because of this wide application range, there is also a large number of different FS hardware realizations, like Phase-Lock-Loops (PLL), Voltage Controlled Oscillators (VCO) [1, 4] or Direct Digital Frequency Synthesizers (DDFS). The use of DDFS has turned out to be an efficient solution regarding power consumption, stability and accuracy [5]. As given in Fig. 1 it mainly consists of two separate sub-units, the phase accumulator and the sine mapper. While the former is easy to realize by common hardware means, for the latter a huge number of approximation approaches have been considered up to now (see Sec. 2). Those approaches mainly reduce the calculation effort at the cost of accuracy, which is estimated by the Spurious Free Dynamic Range (SFDR) for DDFS designs [6]. In detail, this method determines sine replication quality considering the harmonic content.

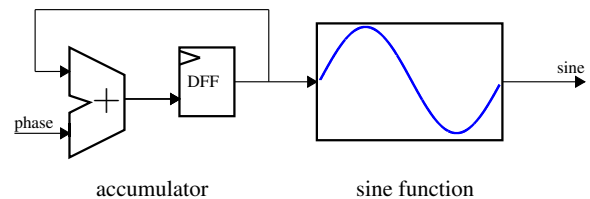


Fig. 1. Architectural overview of a common DDFS hardware structure.

In this paper we propose a novel DDFS design approach which uses an Automatic Nonuniform Piecewise function Approximation (ANPA). With this method quick and versatile DDFS design is possible regarding accuracy and complexity. Basically, the sine generation is realized by Piecewise linear Approximation (PA) considering both hardware efficient function segmentation and high-performance signal processing. Moreover, the nonuniform function approximation is performed automatically by specifying design constraints in advance. As this enables fast generation of multiple different DDFS configurations, easy application specific design is possible. In the following Section an overview about recent DDFS approaches is given. Section 3 is primarily concerned with the main idea of ANPA-based function generation for DDFS. Next, the results of MATLAB simulation, FPGA emulation and IC design are presented (Sec. 4) before our work is concluded in the last Section.

2. RELATED WORK

In the area of PA-based DDFS design, a large number of approaches has been presented so far, mostly based on linear regression as underlying principle. Thus, polynomial, quadratic, linear and approximations were for instance considered, e.g. by [7], [8] and [9], respectively. Recently, De Caro et al. presented a hand-optimized nonuniform approach with three fixed segmentation schemes and continuous function slopes [10]. Further efficiency increase has been reached

by a multiplier-less gradient calculation. Their evaluation proved the nonuniform piecewise approximation methodology to be of high potential and a promising approach for the DDFS generation.

3. ANPA-BASED DDFS DESIGN

In order to improve hand-optimized nonuniform DDFS designs, the ANPA approach enables the handling of more complex structures. In detail, only a small set of segments can be regarded for efficient manual nonuniform PA, which will cause a more complex signal processing effort. In contrast, ANPA enables the use of a large amount of segments which may lead to an optimized data path with a very small number of adder or subtracter units (see Sec. 4). In general, two main aspects can be identified for our DDFS approximation method: on the one hand the ANPA heuristic and on the other hand the HDL generation. Thus, both aspects are described in detail in the following.

3.1. Nonuniform PA

The main principle of nonuniform PA can be seen as an expansion of common equidistant PA. Thus, the original function is split up into several sub-function segments possessing a linear approximation equivalent. Each of these representatives can be expressed in mathematical terms as

$$f(x) = \alpha_0 x + \beta_0, \quad (1)$$

where x denotes the input data, $f(x)$ the corresponding function, α_0 the gradient and β_0 the function offset. In order to achieve high hardware efficiency, some special means must be considered. On the one hand, the input range of the regarded function is set to

$$B - A = 2^{h_{max}}, h_{max} \in \mathbb{N}^+, \quad (2)$$

where A and B are start and end point of the function, respectively. On the other hand, the range of each segment is given as

$$\text{seg}(i) = \text{seg}(i-1) + \frac{B-A}{2^{h_i}}, h_i \in \mathbb{N} \quad (3)$$

with h_i as size coefficient of the i th interval and $\text{seg}(0) = A$. In equidistant PA case the interval lengths are equal for all segments ($h_i = h_0$). Thus, segmentation can be performed regarding the Most Significant Bits (MSB) of the input data. For example $h_0 = 3$ delivers eight equidistant segments with linear equations each. For nonuniform PA, the size of a single segment may vary as it is described in [10]. Note, that this also causes a varying number of MSB that must be regarded for each segment. A typical hardware realization is given in Fig. 2a.

For further computation effort reduction, multiplier-less gradient calculation is applied [11]. In detail, the multiplication is replaced by a limited set of shift operator superpositions. This method can be interpreted either in mathematical terms as a gradient quantization or, from a digital design point of view, as a simplified tree-multiplier structure, where only

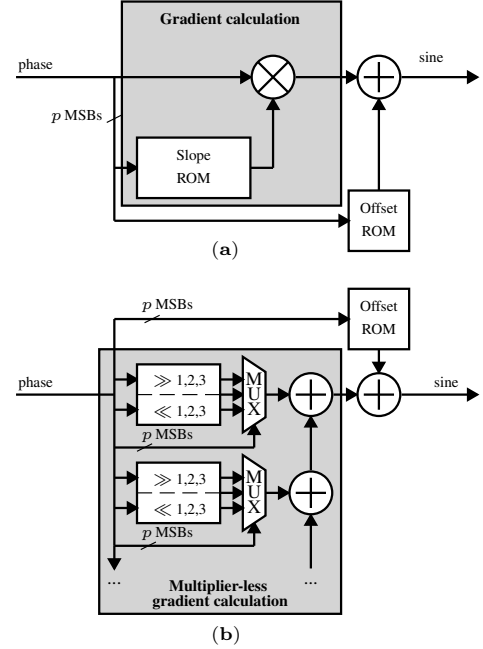


Fig. 2. Overview of (a) common PA and (b) multiplier-less PA hardware structure.

a small set of partial products is considered. As single partial products can be achieved by simple hardware shifts, the calculation effort can be controlled by limiting the total number of superpositions. As a feature, subtraction of two partial products is also enabled. With these modifications, our proposed function approximation can be expressed

$$\tilde{f}(x) = \left[\sum_{j=0}^{n-1} \pm \begin{pmatrix} 2^{\lambda_{0,j}} \\ 2^{\lambda_{1,j}} \\ \dots \\ 2^{\lambda_{m-1,j}} \end{pmatrix} x + \beta \right] \cdot \kappa(x); \lambda_{i,j} \in \mathbb{Z}, \quad (4)$$

with λ determining the actual partial product, m as number of segments and i, j as segment and partial product indexes, respectively. Moreover, n is the Quantization Factor (QF) which is equal to the total number of partial products. The $\kappa(x)$ vector determines the actual linear equation

$$\kappa(x) = \begin{cases} (1, 0, \dots, 0)^T; & A \leq x < \text{seg}(1) \\ (0, 1, \dots, 0)^T; & \text{seg}(1) \leq x < \text{seg}(2) \\ \dots & \\ (0, 0, \dots, 1)^T; & \text{seg}(m-1) \leq x < B \end{cases}. \quad (5)$$

A graphical overview of the corresponding digital architecture is given in Fig. 2b.

3.2. Automatic HDL generation

The automatic HDL generation concentrates on hardware creation by specifying several design parameters. In detail, the

average accuracy, the number of superposition stages for gradient approximation and the resolution are required. Additionally, the entire computation can be separated into two different steps, the linear coefficient estimation with function segmentation and the HDL code generation, presented in the following. The main idea of our automatic HDL generation refers to an accuracy-driven approximation approach. Thus, the segmentation is performed automatically considering the average accuracy specified in advance. By comparison of the actual approximation results, segment bisection is performed if necessary. Due to (2), both new produced segments fulfill segment equation (3). The entire automatic HDL generation heuristic is implemented in Matlab. In addition, the linear equation estimation is performed straight forward from the lower border A to the upper border B. All possible quantized gradients are compared to the results of an unmodified multiplication, what from the best approximation is selected. For function offset estimation, again the common linear approximation is regarded. In detail, it is achieved by determining the smallest sum of absolute errors for both functions. Algorithm 1 shows a pseudo-code description of the entire heuristic. After the function approximation is completed, the relevant data is extracted and translated to corresponding HDL code. In order to enable an easy and flexible output file generation, string templates are used. Considering additional parameters, such as data path width or number format, the estimated linear coefficients are written automatically to a Verilog file. To keep the translation effort as little as possible, the coefficients are mapped directly to appropriate hardware structures inside the string templates. E.g. for the segment selection, multiplexer units are used and the multiplier quantization is realized by adder and subtracter units.

4. RESULTS

For ANPA-based DDFS quarter sine functions are approximated considering the input data scaling mentioned in Sec. 2. In order to obtain the full sine wave, function flipping is processed as e.g. described in [9]. For a detailed evaluation, simulation, FPGA emulation and IC synthesis is performed in this work. As for full DDFS functionality also an accumulator is required, this is also installed. All results are described in detail in the following.

4.1. Simulation

In order to find appropriate hardware implementation candidates, exhaustive simulation is performed. Thus, several different (single phase) ANPA DDFS configurations are evaluated considering complexity and accuracy. For the former, gradient QF 1-3 are investigated, as they correspond directly to adder or subtracter units. The latter is qualified, regarding the previously introduced SFDR and using resolutions of 12, 16 and 20 bit. In order to enable fair comparison of different QFs, similar SDFR values are compared. As the bit resolution has a huge impact on the SFDR, these borders are set up inde-

Algorithm 1: Automatic DDFS generation.

```

1 // Input: quarter sine function  $f = f(x)$ ,
2 // resolution  $r$ , quantization factor  $q$ ,
3 // average accuracy  $a$ 
4 int  $i_{max}, i_c = 1$ ;
5 int[]  $f_c = f$ ;
6 /// runtime
7 do {
8    $m_c[i] = \text{estimateQuantizedGradient}(f_c, r, q)$ ;
9    $b_c[i] = \text{estimateBias}(m_c, f_c)$ ;
10  // accumulate average error
11  for ( $j=1; j < \text{getLength}(f_c); j++$ ) {
12     $\text{abs\_err} = \text{abs}((m_c[i] \cdot j + b_c[i]) - f_c[j])$ ;
13  }
14   $\text{avg\_err} = \text{abs\_err} / \text{getLength}(f_c) - 1$ ;
15  // perform bisection or move on to next segment
16  if ( $\text{avg\_err} > a$ ) {
17     $f_c = \text{getSubvector}(f_c, 0, \text{getLength}(f_c)/2 - 1)$ ;
18     $i_{max} + +$ ;
19  } else {
20     $f_c = \text{nextSegment}(f)$ ;
21     $i + +$ ;
22  }
23 }
24 while ( $i_c < i_{max}$ );
```

pendently, considering the respective maximum SFDR. The entire simulation results are given in Fig 3. For 12 bit resolution, a maximum SFDR of 65dBc is reached. Although this value is not sufficient for many applications, only small hardware effort is necessary. Thus, even a QF of 1 gives satisfying results with an acceptable number of segments. As a curiosity, regarding lower accuracy, an equal number of segments is required for a varying SFDR. This can be explained by the straight forward approach concentrating on optimization in the actual segment only. The 16 bit configuration allows an SFDR increase to 90dBc. While for QFs of 2 and 3 the number of segments rises only slowly, 834 segments are required for QF 1. For the last configuration (20 bit resolution), 110dBc SFDR are reached. But in this case, a total number of 8939 and 988 segments is used for 1 and 2 gradient quantization, respectively, which cause these approaches to be completely unusable. Only for factor 3 tolerable results are achieved. In summary, it can be stated that varying the resolution allows several different DDFS designs. While for small bit widths a gradient quantization of 1 delivers sufficient results, both must be increased for higher accuracy. For hardware implementation, QFs of 1, 2 and 3 are chosen for an SFDR of

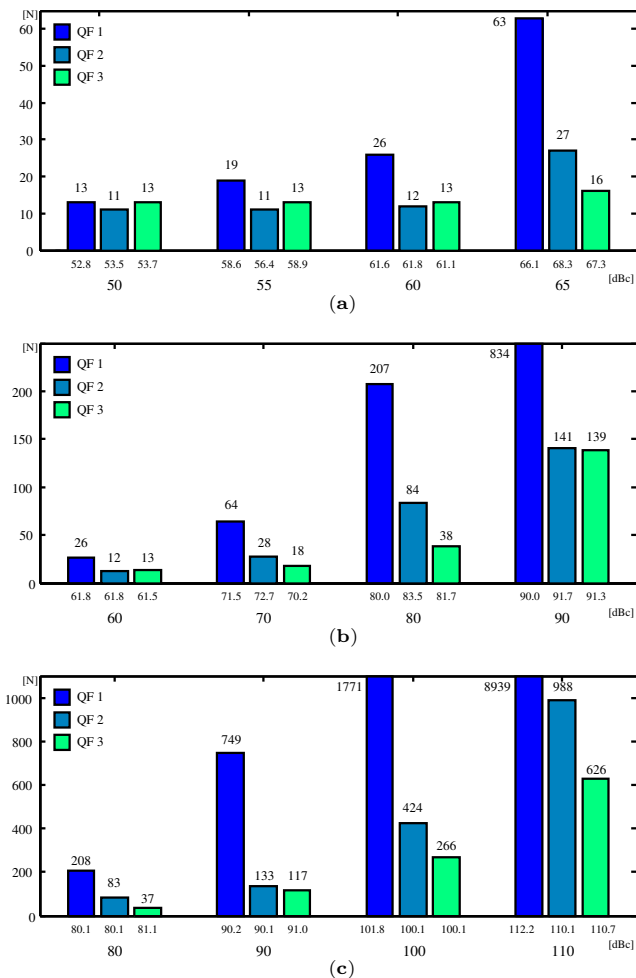


Fig. 3. Number of segments over SFDR for ANPA-based quarter sine function approximation with quantization factors (QF) 1, 2 and 3 as well as (a) 12bit, (b) 16bit and (c) 20bit resolution. The first row under the bars shows the real SFDR

60dBc, 90dBc and 110dBc with a bit width of 12, 16 and 20 bit, respectively (ANPA₆₀, ANPA₉₀ and ANPA₁₁₀).

4.2. FPGA emulation

For the FPGA emulation, a multiprocessor board is used, equipped with a DSP and Virtex 4 FPGA. The entire synthesis is performed by the XILINX ISE(tm) toolchain. Thus, the standard sine function is implemented in C on the DSP. The three chosen ANPA approaches are mapped on the FPGA and controlled by the DSP. In detail, input data and results are sent to the FPGA and compared, respectively. For each configuration, exhaustive verification is performed as well as the average accuracy is verified. An overview of the synthesis results is given in Tab. 1.

Table 1. FPGA synthesis results of the three ANPA hardware implementations.

Virtex4 XC4	ANPA ₆₀	ANPA ₉₀	ANPA ₁₁₀
Number of Slices	85	301	1252
Number of Registers	15	25	36
Frequency [MHz]	352	344	322

4.3. IC design

For IC design, logical and physical synthesis is performed using UMC-Faraday 130nm technology with a supply voltage of 1.2V. In order to obtain meaningful results, timing back annotation is included as well as parasitic effects are considered. Also balanced synthesis configuration is chosen.

The synthesis of the ANPA₆₀ approach delivers a frequency of 389MHz, a power consumption of 2.87 μ W/MHz and a core area of 4191 μ m². For ANPA₉₀ the frequency decreases to 278MHz compared to ANPA₆₀, which is explainable by the two adder units used for phase to sine mapping. The power consumption increases to 5.53 μ W/MHz as well as the area (7396 μ m²). As ANPA₁₁₀ requires 3 adders for sine realization, it only reaches 133MHz, 24.80 μ W/MHz and 28900 μ m² for frequency, power consumption and area. In a next step, the synthesis results are compared to actual reference designs with a similar SFDR. Note, that in order to allow the comparison of implementations with a different underlying technology size, the normalized area is used here [9]. Regarding an SFDR of 60dBc and 90dBc, it turns out ANPA achieves best results regarding power efficiency and frequency. However, area results are worse than actual references, due to the large amount of nets and the higher bit width used in our proposal. A complete overview is given in Tab 2.

5. CONCLUSION AND FUTURE WORK

In this paper a new approach for Direct Digital Frequency Synthesis (DDFS) is introduced, which focuses on Automatic Nonuniform Piecewise linear function Approximation (ANPA). Effective linear approximation of the required sine function implementation is installed. Multiplier-less gradient calculation is performed considering the specified accuracy. For insufficient results, bisection delivers smaller segments; this mostly leads to better approximation as well as hardware efficient segment access is still enabled. The entire heuristic delivers a efficient DDFS design considering complexity, power and timing. The automated realization enables the investigation of a large set of nonuniform DDFS design configurations. Thus, several different approaches are generated, regarding its Spurious Free Dynamic Range (SFDR). Three different configurations are selected and mapped on a FPGA platform for verification. Also IC synthesis is performed, which approved ANPA-based DDFS designs as powerful and

Table 2. Comparison of IC synthesis results to actual references.

Reference	Configuration	SFDR [dBc]	Technology [nm]	Quantization Factor	Segments	Frequency [MHz]	Norm. Area [10 ⁵]	Power [μ W/MHz]
This work	ANPA₆₀	61.6	130	1	26	389	2.58	2.87
[10]	nonuniform	62.0	130	3	9	335	2.12	3.16
[9]	quasi-linear	63.2	130	4	4	313	2.22	4.9
[12]	ROM	60.1	250	-	-	334	28.8	25.5
This work	ANPA₉₀	91.7	130	2	141	278	4.38	5.53
[10]	nonuniform	90.3	130	3	32	216	4.21	5.98
[9]	quasi-linear	89.0	130	4	16	178	5.20	13.8
This work	ANPA₁₁₀	110.7	130	3	626	133	17.10	24.89
[12]	ROM	101	250	-	-	201	5.7	61.7

efficient solution. For future work, the automatic function generation must be improved. As shown in the simulation results, an equal amount of segments is achieved for a varying SFDR. This is explainable by the straight forward processing of the presented heuristic. Also, only synthesis with balanced constraint configuration is regarded here. Considering common hardware techniques like pipelining or operand isolation, this may lead to more efficient high-performance designs.

6. REFERENCES

- [1] G. Bischof, B. Scholnick, and E. Salman, "Fully integrated PLL based clock generator for implantable biomedical applications," in *The Annual Conference on Long Island Systems, Applications and Technology (LISAT)*, May 2011, pp. 1–6.
- [2] Wen Fan and Chiu-Sing Choy, "Power efficient and high speed frequency synchronizer design for MB-OFDM UWB," in *IEEE International Conference on Ultra-Wideband, 2009. (ICUWB)*, Sept. 2009, pp. 669–673.
- [3] T.J. Endres, R.B. Hall, and A.M. Lopez, "Design and analysis methods of a DDS-based synthesizer for military spaceborne applications," in *Proceedings of the 1994 IEEE International Frequency Control Symposium*, Jun 1994, pp. 624–632.
- [4] Yoohwan Kim, Byunghak Cho, and Yoosam Na, "A design of fractional-N frequency synthesizer with quad-band (700 MHz/AWS/2100 MHz/2600 MHz) VCO for LTE application in 65 nm CMOS process," in *Asia Pacific Microwave Conference, APMC 2009.*, Dec. 2009.
- [5] Paul Kern, "Direct digital synthesis enables digital PLLs," *RF design*, 2007.
- [6] A. Torosyan and A.N. Willson, "Exact analysis of DDS spurs and SNR due to phase truncation and arbitrary phase-to-amplitude errors," in *Proceedings of the 2005 IEEE International Frequency Control Symposium and Exposition.*, Aug. 2005, p. 9 pp.
- [7] W. Akram and E.E. Swartzlander Jr., "Direct digital frequency synthesis using piece-wise polynomial approximation," in *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, Nov. 2003.
- [8] F. Curticapean and J. Niittylahti, "Direct digital frequency synthesizers of high spectral purity based on quadratic approximation," in *9th International Conference on Electronics, Circuits and Systems*, 2002, vol. 3, pp. 1095–1098.
- [9] A. Ashrafi, R. Adhami, and A. Milenkovic, "A direct digital frequency synthesizer based on the quasi-linear interpolation method," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 4, April 2010.
- [10] D. De Caro, N. Petra, and A.G.M. Strollo, "Direct digital frequency synthesizer using nonuniform piecewise-linear approximation," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 10, pp. 2409–2419, Oct. 2011.
- [11] O. Gustafsson and K. Johanson, "Multiplierless piecewise linear approximation of elementary functions," in *Fortieth Asilomar Conference on Signals, Systems and Computers (ACSSC '06)*, 29 2006–Nov. 1 2006, p. 16781681.
- [12] D. De Caro, N. Petra, and A. Strollo, "Reducing lookup-table size in direct digital frequency synthesizers using optimized multipartite table method," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 7, pp. 2116–2127, Aug. 2008.