Diophantine Frequency Synthesis: A Forward Two-PLL Architecture Case-Study

Paul Peter Sotiriadis

***Abstract*— Diophantine Frequency Synthesis[1] [2] (DFS) is a new approach to high-resolution[3], fast-hopping, low-spurs frequency synthesis. It is based on mathematical properties of integer numbers and employs two or more Phase-Locked Loops (PLL)s to achieve frequency steps** *arbitrary finer* **than the phase-comparator frequencies of the constituent PLLs. A case study of a forward, two-PLL DFS synthesizer is presented that achieves frequency resolution about** 100 **times finer than that of the two constituent PLLs. Spectral measurements demonstrate the resolution of the architecture and the spectral purity of the output signal.**

I. Introduction

Frequency synthesis is involved in many engineering systems. Depending on the application, certain specifications of the synthesizer are more challenging to achieve than others, e.g., fast frequency hopping is needed in high-speed frequency-domain multiplexing and certain ultra-wide-band communication systems, low spurious levels are required in instrumentation systems, very high frequency resolution is important in timing systems and atomic clocks [1]-[2].

Diophantine Frequency Synthesis[1] (DFS) [3]-[4] is a new methodology for designing frequency synthesis architectures having fast frequency hopping, very high resolution (small frequency step) and low spurs[4], at the same time, with low circuit complexity and modular designs.

DFS is based on mathematical properties of integer numbers and Diophantine equations [5]. DFS architectures employ two or more basic phase locked loops (PLL)s whose output frequencies are added (or subtracted) to give the output frequency of the synthesizer.

DFS *distributes* the frequency resolution among the PLLs. Some basic facts of DFS theory are presented in the following section. More details are available in [3].

The paper presents a forward two-PLL DFS architecture and a circuit implementation of it. It illustrates how the high-level DFS methodology can be realized using commercially available PLL devices to achieve a frequency synthesizer with much higher resolution than those of the constituent PLLs without sacrificing hopping speed or PLL loop bandwidths.

Specifically, the output-frequency steps (resolutions)[5] of the two PPLs are $43\,kHz$ and $67\,kHz$ while that of the DFS synthesizer is $542\,Hz$, i.e. about $\times 100$ times finer. The (used) frequency range of the PLLs is $10\,MHz$ and the output frequency range of the synthesizer is $0 - 10\,MHz$.

II. Elements of DFS

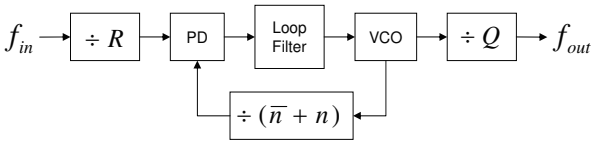

Fig. 1. Basic PLL

DFS uses two or more basic PLLs like that in Fig. 1[6] that have fixed *prescaler* divider R. The feedback divider is the sum $\bar{n} + n$, of a fixed value $\bar{n}$, and, a variable n which can take both *negative* and *positive* values within a predefined range. For all values of n, the sum $\bar{n} + n$ is positive. An output divider, Q, may be present and the output frequency of the PLL is

$$f_{out} = \frac{\bar{n} + n}{Q \times R} f_{in}.$$

The general high-level k-PLL DFS architecture is shown in Fig. 2. The PLLs, with parameters Q, R_i, n_i and $\bar{n}_i$, $i = 1, 2, \ldots, k$, are driven by the same signal and their output frequencies are added (or subtracted) to provide the output frequency of the synthesizer,

$$\begin{aligned} f_{out} = & \left(\frac{n_1}{R_1} + \frac{n_2}{R_2} + \ldots + \frac{n_k}{R_k} \right) \frac{f_{in}}{Q} \\ & + \left(\frac{\bar{n}_1}{R_1} + \frac{\bar{n}_2}{R_2} + \ldots + \frac{\bar{n}_k}{R_k} \right) \frac{f_{in}}{Q} \end{aligned} \quad (1)$$

[1] Patent Pending

[2] The detailed theory of DFS is available at [3]

[3] It can easily achieve sub-Hertz frequency resolution.

[4] ...especially in the vicinity of the carrier - in contrast to methodologies based on dither, interpolation or pulse removal.

[5] The phase-comparator frequencies of the PLLs are $130\,kHz$ and $133\,kHz$ respectively which are different than their output frequency steps because of the additional output dividers, see Fig. 3. Note however that the output dividers are not essential for the DFS.

[6] The Phase Comparator (PC) may also be a Phase-Frequency comparator or detector etc. Also, an output divider (Q) may be present.

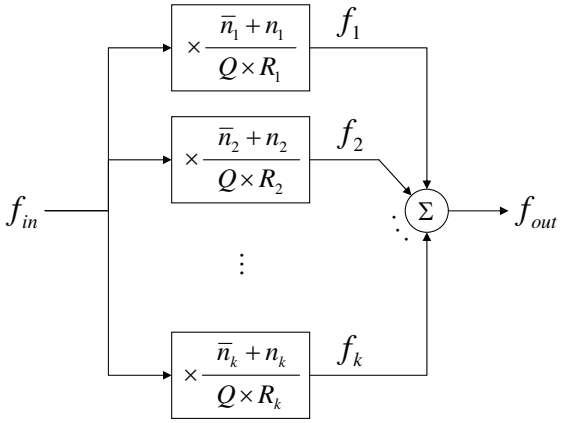

Fig. 2. k-PLLs DFS scheme. Parameters $\bar{n}_1$ to $\bar{n}_k$ are omitted.

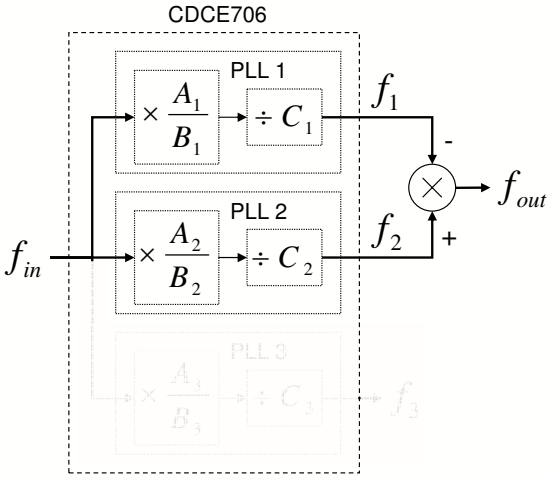

Fig. 3. High-level two-PLL DFS architecture

The second term in the right side of (1) is fixed; moreover we set

$$\bar{f}_{out} = \left(\frac{\bar{n}_1}{R_1} + \frac{\bar{n}_2}{R_2} + \ldots + \frac{\bar{n}_k}{R_k} \right) \frac{f_{in}}{Q} \tag{2}$$

Now we focus on the first term in the right side of (1), which is variable. DFS' main Theorem follows [3].

Theorem 2.1: [3] If $R_1, R_2, \ldots, R_k$ are pairwise relatively prime positive integers (i.e., no pair of them has common divider other than ± 1) then, for every integer n, such that $-R_1 R_2 \cdots R_k \leq n \leq R_1 R_2 \cdots R_k$ we can find $n_1, n_2, \ldots, n_k$ with $-R_i \leq n_i \leq R_i$, for all $i = 1, 2, \ldots, k$, for which

$$\frac{n_1}{R_1} + \frac{n_2}{R_2} + \ldots + \frac{n_k}{R_k} = \frac{n}{R_1 R_2 \cdots R_k}. \tag{3}$$

Rephrasing Theorem 2.1 and using equation (1) we conclude that by appropriately adjusting the feedback dividers, n_i's (and therefore the PLL frequencies), the output frequency, f_{out}, can take all values within the range

$$\left[\; \bar{f}_{out} - \frac{f_{in}}{Q} \;\; , \;\; \bar{f}_{out} + \frac{f_{in}}{Q} \; \right]$$

with *uniform* frequency step equal to

$$f_{step} = \frac{f_{in}}{Q \, R_1 R_2 \cdots R_k}.$$

Note that if some of the PLL frequencies, $f_1, f_2, \ldots, f_k$ were subtracted, instead of being added to the general sum, i.e. if n_i/R_i were replaced by $-n_i/R_i$ for some indices i in (3), the length of the output frequency range would also be $2f_{in}/Q$, i.e. exactly the same as before, because the ranges of $n_1, n_2, \ldots, n_k$ are symmetric with respect to 0. The output frequency step, f_{step}, would also remain the same.

A significant advantage of DFS is that with small values of k, $R_1, R_2, \ldots, R_k$ and Q, the frequency step, f_{step}, can be made very small, while at the same time, the phase-comparator frequencies of the PLLS, i.e. f_{in}/R_i, $i = 1, 2, \ldots, k$, and the output frequency range are large.

III. A Two-PLL Forward-DFS Synthesizer

The purpose of this paper is to present a DFS synthesis example and its associated tricks and not to repeat well-know facts regarding PLL design. To this end off-the-shelf components were used. Specifically, Texas Instruments' programmable triple-PLL CDCE706 device was selected for convenience since it automatically adjusts its loop-filter parameters and VCO's tuning ranges. It is noted however that this design convenience is paid by low SFDR in the order of 50 to $55\,dBc$.

Two of the three PLLs in CDCE706 were employed in the DFS scheme. The high level architecture is shown in Fig. 3. The range of the prescaler, feedback and output dividers are shown in Table I.

Divider	Min	Max
A_i	1	4095
B_i	1	511
C_i	1	127

TABLE I
Ranges of the Frequency Dividers

The total range of the VCO's is $80\ MHz - 300\ MHz$ therefore implying (see Fig. 3) that

$$80\ MHz \leq \frac{A_i}{B_i} f_{in} \leq 300\ MHz.$$

A. Choosing the DFS Parameters

The desirable in this case study has been to achieve frequency range $0 - 10MHz$ with high frequency resolution while maintaining relatively high phase-comparator frequencies in the individual PLLs.

From Fig. 3 we have that, for each PLL $i = 1, 2$,

$$f_i = \frac{A_i}{B_i C_i} f_{in}, \tag{4}$$

the VCO's frequencies are

$$f_{VCO_i} = \frac{A_i}{B_i} f_{in} \tag{5}$$

and the phase-comparator frequencies are

$$f_{PC_i} = \frac{f_{in}}{B_i} \tag{6}$$

Now the frequency dividers in expressions (4) can be written as

$$B_i C_i = R_i Q, \tag{7}$$

$i = 1, 2$, where $Q = \gcd(B_1 C_1, B_2 C_2)$ implying $\gcd(R_1, R_2) = 1$, i.e. R_1, R_2 are relatively prime as it is required by Theorem 2.1. Moreover, we can express $A_i = n_i + \bar{n}_i$ (and impose the constraint required from the Theorem, that $-R_i \leq n_i \leq R_i$) so from (4) we have

$$f_i = \frac{n_i + \bar{n}_i}{R_i} \left(\frac{f_{in}}{Q} \right) \tag{8}$$

From expressions (8), (3), the fact that we have chosen $f_{out} \triangleq f_2 - f_1$ (see Fig. 3) and Theorem 2.1 we have

$$f_{out} = \left(\frac{\bar{n}_2}{R_2} - \frac{\bar{n}_1}{R_1} \right) \frac{f_{in}}{Q} + \frac{n}{R_1 R_2} \frac{f_{in}}{Q} \tag{9}$$

with n ranging from $-R_1 R_2$ to $R_1 R_2$. Therefore f_{out} ranges from $\bar{f}_2 - \bar{f}_1 - f_{in}/Q$ to $\bar{f}_2 - \bar{f}_1 + f_{in}/Q$.

Since the desirable output range of the synthesizer is $0 - 10MHz$, this means that $\bar{f}_2 - \bar{f}_1 - f_{in}/Q = 0$ and $\bar{f}_2 - \bar{f}_1 + f_{in}/Q = 10MHz$, so

$$\bar{f}_2 - \bar{f}_1 = 5MHz \tag{10}$$

and

$$\frac{f_{in}}{Q} \gtrapprox 5MHz \tag{11}$$

The input (reference) frequency used was $f_{in} = 64MHz$, and so $f_{in}/5MHz = 12.8$. Therefore *the maximum value of Q resulting in at least $10MHz$ of output range is*

$$Q = 12 \tag{12}$$

The maximum desirable difference $f_{out} = f_2 - f_1$ is $10MHz$ and in order to minimize spurious signals generated in the mixing, in general, we would like to have high ratio $\bar{f}_i/10MHz$, $i = 1, 2$. However high $\bar{f}_i$ frequencies may also result in relatively high phase noise of f_{out}. A conservative choice is to pick $\bar{f}_1, \bar{f}_2$ close to $100MHz$ so that $\bar{f}_i/10MHz$ is about ten.

Given the frequency limits of the VCO's, this choice allows for C_i's to take any of the three values $\{1, 2, 3\}$. However, *we don't want both of them to take the maximum possible value*, because if $C_1 = C_2 = 3$ then they contribute only a factor of 3 to the product $R_1 R_2 Q$ and this "improves" frequency resolution (see Eq. (9)) only by 3. In contrast, we set

$$C_1 = 3 \text{ and } C_2 = 2 \tag{13}$$

which contribute[7] a factor of 6 to $R_1 R_2 Q$.

To simplify the rest of the derivation we assume that C_1 and C_2 are factors of R_1 and R_2 respectively, i.e. equivalently that the greatest common divider, Q, of $B_1 C_1$ and $B_2 C_2$ is entirely a factor of B_1 and B_2 and so because of (12) it is

$$B_1 = 12X \text{ and } B_2 = 12Y \tag{14}$$

for some integers X and Y. Note however that from Table I we have

$$X, Y \leq 511/12 = 42.583\ldots. \tag{15}$$

To maximize the product $R_1 R_2 Q$, and therefore the resolution of the synthesizer, we could try to give X and Y their maximum possible values, i.e. $X = 42$ and $Y = 41$, or, $X = 41$ and $Y = 42$.

In the first case we get $B_1 C_1 = 1512$ and $B_2 C_2 = 984$ which imply $Q = gcd(1512, 984) = 24$ (which is different than the value of Q we have set), $R_1 = 63$ and $R_2 = 41$ and so $R_1 R_2 Q = 61,992$; in the second case we get $R_1 R_2 Q = 41,328$.

In contrast if we choose $X = 41$ and $Y = 40$, i.e. slightly smaller values, we end up with $Q = 12$, $R_1 = 123$ and $R_2 = 80$ giving $R_1 R_2 Q = 118,080$. We finally get that

$$B_1 = 12 \cdot 41 = 492 \text{ and } B_2 = 12 \cdot 40 = 480 \tag{16}$$

The following set of equations summarizes the derivations up to now:

$$\begin{aligned} B_1 = 492 \quad C_1 = 3 \quad R_1 = 123 \\ B_2 = 480 \quad C_2 = 2 \quad R_2 = 80 \end{aligned}$$

Now, following the previous discussion on $\bar{f}_1$ and $\bar{f}_2$ we choose

$$\bar{n}_1 = 2168 \text{ and } \bar{n}_2 = 1485 \tag{17}$$

which give

$$\bar{f}_1 = \frac{\bar{n}_1}{B_1 C_1} f_{in} = \frac{2168}{492 \cdot 3} 64\, MHz = 94.005,420\, Hz$$

and

$$\bar{f}_2 = \frac{\bar{n}_2}{B_2 C_2} f_{in} = \frac{1485}{480 \cdot 2} 64\, MHz = 99.000,000\, Hz.$$

[7] One has to decide how to "form" the greatest common divider of the pair $R_1 Q_1$ and $R_2 Q_2$...

Finally, following the DFS guidelines, the ranges of the dividers A_1 and A_2 are

$$\begin{array}{rcccl} 2168-123 & \leq & A_1 & \leq & 2168+123 \\ 1485-80 & \leq & A_2 & \leq & 1485+80 \end{array} \quad (18)$$

The above choices of the prescalers and output dividers and the ranges of the feedback dividers result in the DFS scheme with frequencies given in Table II.

B. The Mixer

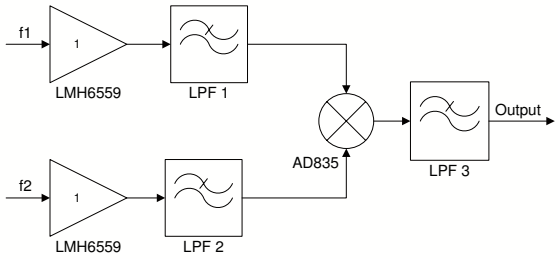

Fig. 4. Mixer, Low-Pass Filters and Buffers

The high-level schematic of the mixer is shown on Figure 4. The mixer is based on the four-quadrant multiplier AD835 chosen for its high linearity and appropriate bandwidth[8]. Two wide-band buffers, LMH6559, are used to isolate the PLLs from the filters and the multiplier and reduce backward signal leakage.

The two low-pass filters LPF1 and LPF2 are 7^{th} order Chebyshev Type-I with passive ladder implementations and $1\,dB$ ripple in the pass-band. They remove the harmonics of the square-wave signals coming from the PLLs and feed clean sinusoidals to the multiplier. The pass-bands of LPF1 and LPF2 are $0-112\,MHz$ and $0-123\,MHz$ respectively.

The third filter, LPF3 following the multiplier removes the high frequency product of the multiplication, at around $200\,MHz$. It is a 5^{th} order low-pass Chebyshev Type-I passive ladder filter with pass-band $0-10\,MHz$ and $1\,dB$ ripple.

IV. Results

Programming the two PLLs using the DFS parameters derived in Section III-A and the DFS algorithms in [3], it is possible to generate all predicted frequencies

$$f_{out} \;=\; 4,994,580\,Hz + n\cdot 542\,Hz \quad (19)$$

with integer n ranging

$$n \;=\; -9840\ldots 9840. \quad (20)$$

Given the desirable value of n, parameters n_1 and n_2 are derived using the theory and algorithms in [3]. The frequency ranges of the PLLs and the output signal, along with their resolutions, are shown in Table II.

	Min	Central	Max	Step
f_{in}	-	64.000,000	-	-
f_1 (PLL 1)	88.672,087	94.005,420	99.338,753	43,360
f_2 (PLL 2)	93.666,667	99.000,000	104.333,333	66,667
f_{out} (DFS)	-338,753	4,994,580	10,327,913	542

TABLE II
Frequency Ranges and Resolutions of the signals in the DFS scheme (in Hz).

Therefore this simple two-PLL DFS scheme achieves frequency step (resolution) of $542\,Hz$ when the output-frequency steps of the constituent PLLs are

$$\begin{aligned} f_{1_{step}} &= 43,360\,Hz \\ f_{2_{step}} &= 66,667\,Hz \end{aligned}$$

In addition, this allows for *fast frequency hopping* and easy removal of the side-band spurs introduced by the phase-comparator[9], leading to clean output spectrum.

Fig. 5 shows the spectrum of the (output) signal for five consecutive synthesized frequencies and demonstrates the $542\ Hz$ resolution.

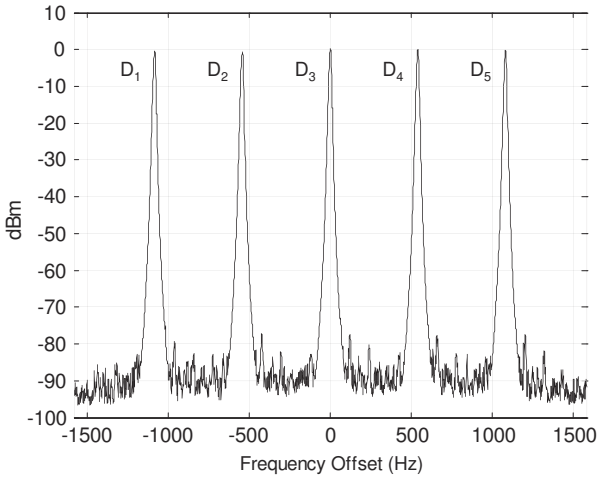

Fig. 5. Five consecutive frequencies, D$_1$: $3,690,515Hz$, D$_2$: $3,691,057Hz$, D$_3$: $3,691,599Hz$, D$_4$: $3,692,141Hz$, D$_5$: $3,692,683Hz$. The DFS frequency step is $542\,Hz$ while the constituent PLLs' frequency steps are $43\,kHz$ and $67\,kHz$. For the measurements it was used: RBW=10Hz, VBW=10Hz.

The parameters of the synthesizer resulted in the frequencies shown in Fig. 5 are listed in Table III.

[8]Although this multiplier can be used as a mixer in several applications, better noise-level performance can be achieved by other types of mixers.

[9]The phase-comparator frequencies of the PLLs are even higher than the corresponding output steps, specifically, $f_{PC_1} = 130,081\,Hz$ and $f_{PC_2} = 133,333\,Hz$.

	n_1	n_2	n	f_1	f_2
D_1	-27	-2	$7,435$	$95,176,151$	$98,866,666$
D_2	-7	-15	$7,436$	$94,308,943$	$98,000,000$
D_3	-110	52	$7,437$	$98,775,067$	$102,466,666$
D_4	-90	39	$7,438$	$97,907,859$	$101,600,000$
D_5	-70	26	$7,439$	$97,040,650$	$100,733,333$

TABLE III

FREQUENCY RANGES AND FREQUENCY STEPS (RESOLUTIONS) OF THE SIGNALS IN THE PRESENTED DFS SCHEME. ALL FREQUENCIES ARE IN HZ.

V. CONCLUSIONS

Application aspects of the Diophantine Frequency Synthesis (DFS) methodology have been presented through the design of a forward two-PLL DFS frequency synthesizer. DFS is based on number theory and Diophantine equations, uses two or more basic PLLs and allows for independent choices for the output frequency step (resolution) and the phase-comparator frequencies of the PLLs. The Diophantine Frequency Synthesis (DFS) leads to very fine frequency resolution and fast frequency hopping architectures with low spurs, especially in the vicinity of the carrier.

REFERENCES

[1] Vadim Manassewitsch, "*Frequency Synthesizers*" , third edition, John Willey & Sons, 1987.

[2] William F. Egan, "*Frequency Synthesis by Phase Lock*", second edition, John Willey & Sons, 1999.

[3] Paul P. Sotiriadis, "Diophantine Frequency Synthesis, *EEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, Vol. 53, No. 11, Nov. 2006, pp. 1988-1998.

[4] Paul P. Sotiriadis, "Diophantine Frequency Synthesis; The Mathematical Principles", *IEEE Internat. Symp. on Circuits and Systems 2007*.

[5] Daniel E. Flath, "*Introduction to Number Theory*", John Willey & Sons, 1989.