Application Aspects of the Diophantine Frequency Synthesis Methodology

Invited Paper

Paul P. Sotiriadis
EMAIL : pps at ieee dot org

***Abstract*— This work considers two aspects of the design of Diophantine Frequency Synthesizers[1] which are important for wireless applications: I) the choice of DFS parameters leading to decimal frequency resolution (10^m Hz) and acceptable input reference frequency, and, II) the choice of the mixer(s) used in DFS architectures and the frequency planning in the constituent PLLs that lead to high Spurious Free Dynamic Range at the output.**

I. Introduction

Frequency synthesis is used in the majority of modern electronic systems. Quality frequency synthesizers are critical components in wireless and wired communications, radars, navigation, instrumentation and in many other systems.

Diophantine Frequency Synthesis (DFS) is a relatively new methodology for frequency synthesis; it was introduced in [1] and was further evolved in [2]. DFS is a high-level approach in the sense that it uses a two or more simpler frequency synthesis blocks, like integer-N PLLs[2], programs their frequency multiplication factors appropriately and adds/subtracts their output frequencies to generate the output frequency of the synthesizer.

Based on properties of the solutions of Diophantine equations and using certain properties of integer numbers, DFS distributes the frequency resolution of the synthesizer among the constituent PLLs. Doing so, it allows for relatively high phase-detector frequencies at the constituent PLLs and therefore fast frequency hopping and suppressed phase-detector spurs. If necessary attention is paid in the design of the mixers, DFS can lead to high resolution, fast hopping synthesizers with high Spurious Free Dynamic Range (SFDR).

Two aspects of DFS application that are especially important for wireless applications are discussed in this work.

The first is the choice of the DFS programming parameters that lead to decimal (i.e. 10^mHz) output frequency step (resolution). Most communication standards require frequency step equal to a power of 10 or a product of it. The same is true for instrumentation and other equipment.

Conditions for decimal frequency steps leading to a balanced distribution of the resolution to the constituent PLLs are presented and the case of the GPRS standard is considered as an example.

[1]Patent Pending

[2]Although practically any standard frequency synthesis block can be used, integer-N PLLs are preferred for convenience in illustrating the method.

Then, the mixing of PLL output frequencies and the choice of the appropriate mixer are discussed. Minimizing mixing spurs is critical in obtaining a clean output signal. It can be very easy or very challenging depending on the desirable specifications, the design constraints and the implementation form.

II. Elements of DFS for 2-PLL Schemes

Although the general DFS theory can be found [1], it is convenient to briefly review some basic facts for the special case of 2-PLL DFS schemes.

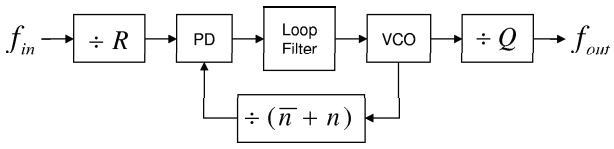

Fig. 1. Basic PLL

DFS employs two or more simple frequency synthesis blocks, like integer-N PLLs shown in Fig. 1 having fixed prescaler divider R and feedback divider that is the sum $\bar{n}+n$, of a fixed value $\bar{n} > R$, and, a variable n which can take values within the range $-R, \ldots, R$. An output divider, Q, may also be present. The output frequency of the PLL is

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

A high-level 2-PLL DFS architecture is shown in Fig. 2. Both PLLs are driven by the same signal and their output frequencies are added (or subtracted) to provide the output frequency of the synthesizer

$$f_{out} = \left(\frac{n_1}{R_1} + \frac{n_2}{R_2}\right)\frac{f_{in}}{Q} + \bar{f}_{out} \tag{1}$$

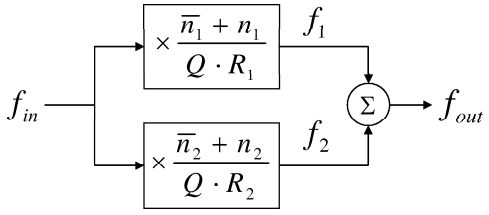

Fig. 2. Abstract 2-PLLs DFS Scheme.

where $\bar{f}_{out}$ is fixed and equals the central output frequency of the synthesizer. It is

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

The following theorem is a special case (for 2 PLLs) of DFS' main theorem [1].

Theorem 2.1: [1] If $\gcd(R_1, R_2)$=1 then, for every integer n, such that $-R_1R_2 \leq n \leq R_1R_2$ we can find integers n_1, n_2 with $-R_i \leq n_i \leq R_i$, $i = 1, 2$, for which

$$\frac{n_1}{R_1} + \frac{n_2}{R_2} = \frac{n}{R_1R_2}. \tag{3}$$

Theorem 2.1 tells us that by appropriately adjusting the feedback dividers, n_1, n_2 within the ranges $-R_1 \leq n_1 \leq R_1$ and $-R_2 \leq n_2 \leq R_2$, respectively, the output frequency can take every one of the values

$$f_{out} = \frac{n}{R_1R_2} \cdot \frac{f_{in}}{Q} + \bar{f}_{out} \tag{4}$$

where $n = -R_1R_2, \ldots, R_1R_2$; i.e. the range of f_{out} is

$$\left[\bar{f}_{out} - \frac{f_{in}}{Q} \ , \ \bar{f}_{out} + \frac{f_{in}}{Q} \right] \tag{5}$$

and the (uniform) frequency step of it is

$$f_{step} = \frac{f_{in}}{Q\, R_1R_2}. \tag{6}$$

Nothing changes if we replace $\frac{n_1}{R_1} + \frac{n_2}{R_2}$ by $\frac{n_1}{R_1} - \frac{n_2}{R_2}$ or $-\frac{n_1}{R_1} + \frac{n_2}{R_2}$ in (3), except the central value $\bar{f}_{out}$ which should be adjusted accordingly. This is true because the ranges of n_1 and n_2 are symmetric with respect to zero. Generalizations of Theorem 2.1 for any number of basic frequency synthesis blocks (PLLs here) can be found in [1] and [2].

III. DFS with Decade-Based Resolution

Many important applications require synthesizers with frequency steps equal to 10^h (Hz), where h is an integer number. From expression (6) this means that

$$f_{step} = \frac{f_{in}}{Q\, R_1R_2} = 10^h \tag{7}$$

It may or may not be possible to satisfy (7) exactly, depending on the constraints of the design and the specific numbers involved.

We consider the case where we are allowed to select the input reference frequency, preferably within a set of "standard" frequencies, and, the common frequency divider in the PLLs is set to one, i.e, $Q = 1$.

Starting with the 2 - PLL DFS scheme in Figure (2) we can choose

$$R_1 = 2^{m_1} \ , \ R_2 = 5^{m_2}. \tag{8}$$

since they lead to "standard" frequencies for several combinations of values of integers m_1, m_2. Since $\gcd(R_1, R_2) = 1$, this choice is acceptable for DFS.

m_1	m_2	R_1	R_2	R_1R_2	$A(m_1, m_2)$
2	1	4	5	20	1.25
3	1	8	5	40	1.60
4	2	16	25	400	1.56
5	2	32	25	800	1.28
6	3	64	125	8000	1.95
7	3	128	125	16000	1.02
8	3	256	125	32000	2.05
9	4	512	625	320000	1.22
10	4	1024	625	640000	1.64
11	5	2048	3125	6400000	1.53
12	5	4096	3125	12800000	1.31
13	6	8192	15625	128000000	1.91
14	6	16384	15625	256000000	1.05

TABLE I
RANGES OF THE FREQUENCY DIVIDERS

Now note that the frequency hopping speed of the synthesizer is essentially equal to the minimum of the frequency hopping speeds of the two PLLs[3]. Moreover, the frequency hopping speed of a PLL is approximately proportional to its loop band-width which is typically approximately proportional to the phase-detector frequency,

$$\begin{aligned} f_{pd_1} &= 10^h R_2 \\ f_{pd_2} &= 10^h R_1 \end{aligned} \tag{9}$$

Based on these, and given that the resolution depends on the product R_1R_2, it makes sense to try to have $R_1 \approxeq R_2$. Formally this can be written as the minimization of

$$A(m_1, m_2) = \frac{\max\{\, 2^{m_1}, 5^{m_2} \}}{\min\{\, 2^{m_1}, 5^{m_2} \}} \tag{10}$$

If we consider m_1 as a free variable and m_2 as a function of m_1, then the minimum of the expression (10) is equal to 1 and is achieved when

$$m_2 = \frac{\ln(2)}{\ln(5)}\, m_1$$

which however is not an integer number for $m_1 \in \mathbb{Z}$. Instead, the optimal integer value of m_2 is given by (11) where $< x >$ is a nearest integer to x.

$$m_2 = \left\langle \frac{\ln(2)}{\ln(5)}\, m_1 \right\rangle. \tag{11}$$

Table I shows the values of m_2 for m_1 ranging from 1 to 14 along with the values of the prescaler dividers R_1, R_2, their products and the corresponding values of A in (10).

Example 3.1: Suppose we would like to have output frequency step equal to 1kHz, which means that $h = 3$, and phase-detector frequencies at the PLLs around 100kHz. Equation (9) implies that we need to find R_1 and R_2 so that $100\text{kHz} \approxeq R_i \cdot 1\text{kHz}$. A convenient choice, from Table I is $R_1 = 2^7 = 128$ and $R_2 = 5^3 = 125$, which because of (7) implies $f_{in} = 16\,\text{MHz}$.

[3]This is because the contribution of the (filters in the) mixer is typical negligible. The only possible exception to this is when $f_{out} << f_1, f_2$.

The ranges of f_1 and f_2 are derived from (5), and, since $Q = 1$, they are

$$f_1 \,:\, \left[\frac{\bar{n}_1}{128} - 1\,, \ldots,\, \frac{\bar{n}_1}{128} + 1\right] \cdot 16\,\text{MHz}$$

and

$$f_2 \,:\, \left[\frac{\bar{n}_2}{125} - 1\,, \ldots,\, \frac{\bar{n}_2}{125} + 1\right] \cdot 16\,\text{MHz}.$$

The values of $\bar{n}_1$ and $\bar{n}_2$ as well as the choice of $f_{out} = f_1 + f_2$ or $f_{out} = f_1 - f_2$ determine the central frequencies $\bar{f}_1$, $\bar{f}_2$ and $\bar{f}_{out}$.

Example 3.2: Now suppose we want to design a DFS synthesizer for the General Packet Radio Service (GPRS) system. GPRS has a 200kHz channel spacing, reverse channel frequencies 890-915 MHz and forward channel frequencies 935-960 MHz.

Lets assume for the moment a 100kHz frequency step requirement, i.e. $h=5$, and that we need to cover both the reverse and forward channels range, i.e. $(960-890)$MHz/200kHz$=$ 350 $(+1)$ synthesized frequencies are needed.

From Theorem 2.1 we know that the number of synthesized frequencies using a 2-PLL DFS scheme is equal to $2R_1R_2+1$. Based on this, it is convenient to select $R_1 = 16$ and $R_2 = 25$ from Table I giving $2R_1R_2 + 1 = 801$.

In this case equation (7) results in $f_{in} = 40$MHz. Taking into account the factor of two in the frequency step we conclude that $f_{in} = 80$MHz. Then, the phase detector frequencies are $f_{pd1} = f_{in}/R_1 = 5$MHz and $f_{pd1} = f_{in}/R_1 = 3.2$MHz.

We can choose the central output frequency to be at the center of the total frequency band, i.e. $\bar{f}_{out} = 925$MHz. Moreover, note that expression (2) for $\bar{f}_{out}$ is similar to that for the variable part of f_{out}. This means that we can choose the value of $\bar{f}_{out}$ with the same resolution f_{step}.

IV. Frequency Mixing in DFS

Since DFS requires frequency addition (or subtraction), frequency mixing is necessary. Careful choice of the central frequencies, $\bar{f}_i$, $i = 1, 2$, the frequency ranges $[\bar{f}_i - f_{in}, \bar{f}_i + f_{in}]$, the shape and amplitude of PLLs' output waveforms entering the mixer, and of course the mixer are important in minimizing the unwanted spurious signal components $mf_1 + nf_2$. The difficulty of these choices varies significantly from one application to another.

It is important to note that frequency mixing in synthesizers is quite different from frequency mixing in RF receivers. In the first case both mixed signals have large amplitudes which means that strong unwanted intermodulation products will be generated if we don't pay the necessary attention to the design; at the same time, noise is not usually an issue. In contrast, in RF receivers, the local oscillator's signal is strong but the input signal is typically very small, this means that unwanted intermodulation products are usually generated only by the harmonics of the local oscillator, and they are far away from the frequency range of interest[4]; moreover, the mixer's noise is a major issue.

[4]... with the exception of Ultra Wide Band systems

Consider the 2-PLL DFS scheme in Figure 2. Assuming for simplicity that $Q = 1$, equation (5) gives that the PLL frequencies f_i, $i = 1, 2$, range within

$$[\bar{f}_i - f_{in}\ ,\ \bar{f}_2 + f_{in}],$$

$i = 1, 2$ respectively in order to generate all output frequencies covering the range

$$[\bar{f}_{out} - f_{in}\ ,\ \bar{f}_{out} + f_{in}]$$

with step equal to $f_{step} = f_{in}/(R_1R_2)$.

If $f_{out} = f_1 - f_2$ the DFS algorithm [1] implies that for all output frequencies the pair (f_1, f_2) lies within the hexagon in Figure 3 below, [2].

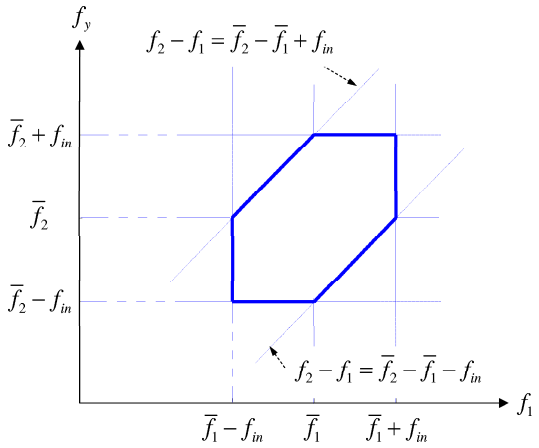

Fig. 3. Ranges of variation of f_1, f_2 and f_{out} when $f_{out} = f_2 - f_1$.

Since $f_{out} = f_1 - f_2 = (f_1 + \Delta f) - (f_2 + \Delta f)$ for every frequency offset Δf, we can shift the hexagon in parallel to the $y = x$ line without changing the output frequency. This gives us some freedom to minimize output spurious signals.

When $f_{out} = f_1 + f_2$ the pair (f_1, f_2) lies within the hexagon in Figure 4.

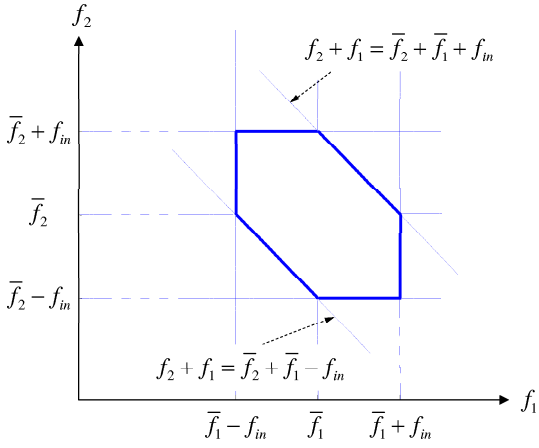

Fig. 4. Ranges of variation of f_1, f_2 and f_{out} when $f_{out} = f_1 + f_2$.

Here, $f_{out} = f_1 + f_2 = (f_1 + \Delta f) + (f_2 - \Delta f)$ for every frequency offset Δf, and so shifting the hexagon in parallel

to the $y = -x$ line does not change the output frequency. As before, we can use this flexibility to minimize output spurious signals.

A. The Choice of the Mixer

Several types of mixers can be used for DFS. A rough classification can be done based on the waveforms of the two inputs (or equivalently the smoothness of the mixer's nonlinearity with respect to each of the input variables).

The "smoothest" mixing is performed by multiplication of sinusoidal signals. As shown in Figure 5 the signals $x_1(t)$ and $x_2(t)$ from the PLLs are filtered and their harmonics are suppress to yield pure sinewaves that enter a multiplier. The multiplier is typically implemented as a Gilbert cell with appropriate input "pre-distortion" so that it does not distort any of the signals [3].

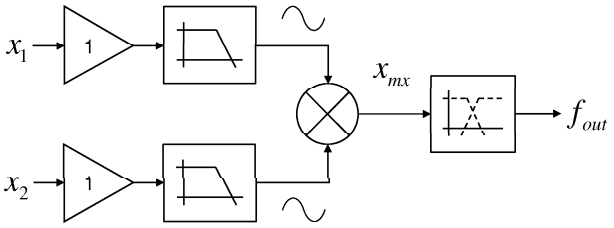

Fig. 5. Mixing of two sinusoidal signals

If $\hat{x}_1(t) = A_1 \cos(w_1 t)$ and $\hat{x}_2(t) = A_2 \cos(w_2 t)$ are the outputs of the filters corresponding to x_1 and x_2 respectively, then the output of the multiplier-mixer is[5]

$$\begin{aligned} x_{mx}(t) \;=\;& A_{mx} \cos\big((w_1 + w_2)t + \phi\big) \\ &+ B_{mx} \cos\big((w_1 - w_2)t + \psi\big) \end{aligned} \tag{12}$$

The filter at the output of the mixer selects one frequency or the other. The filter may be low-pass, band-pass, high-pass or even band-reject.

One can use a *quadrature* version of the multiplier-mixer to remove the unwanted term from (12) and easy the requirements for the output filtering. To use the quadrature mixer we need to generate the I and Q signal components of the two PLLs. Perhaps the easiest way to do this in DFS is by using quadrature oscillators in the PLLs, or divide the oscillator's frequency by 2 using a divider with quadrature output.

Finally, the use of buffers at the inputs of the mixer helps isolating the PLLs. This is important especially when the phase-detector frequencies of the PLLS, f_{in}/R_i, are close to each other and any coupling can potentially generate beating tones, appearing as output spurs at the intermodulation frequencies of the phase-detector frequencies.

A simpler mixer structure, and perhaps the most common one, that can be used in DFS as well is shown in Figure 6. Here the mixer is essentially a switch multiplying a sinewave with a square-wave. Balanced RF and microwave Diode mixers behave very much like this.

[5]... at least ideally. Also note that the amplitudes and the phases may differ due to bandwidth limitations of the multiplier.

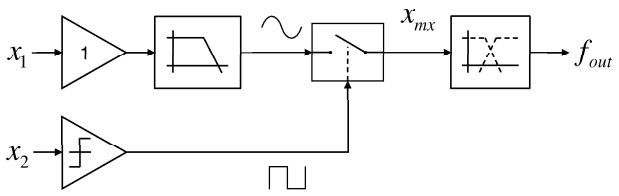

Fig. 6. Mixing of sinusoidal with a square signal

Typically, the switch is implemented differentially and the output signal, x_{mx}, is the multiplication of a the sinewave $\hat{x}_1(t) = A_1 \cos(w_1 t)$ times the ± 1 square-wave. Expressing the last one in Fourier series,

$$\hat{x}_2(t) = \frac{4}{\pi} \sum_{r=0}^{\infty} \frac{1}{2r+1} \sin\big((2r+1)\, w_2\, t\big) \tag{13}$$

we derive the signal at the output of the switch[6]

$$\begin{aligned} x_{mx}(t) = \frac{2A_1}{\pi} \Bigg[& \sum_{r=0}^{\infty} \frac{1}{2r+1} \sin\Big(\big((2r+1)\, w_2 + w_1\big)t\Big) \\ & + \sum_{r=0}^{\infty} \frac{1}{2r+1} \sin\Big(\big((2r+1)\, w_2 - w_1\big)t\Big) \Bigg] \end{aligned} \tag{14}$$

Therefore the output of the switch contains signal components at frequencies

$$w_m^{\pm} = \big| w_1 \pm (2m+1) w_2 \big| \tag{15}$$

with amplitudes[7]

$$A_m = \frac{A_{w_1 \pm w_2}}{2m+1} \tag{16}$$

where $m = 0, 1, 2, \ldots$, and $A_{w_1 \pm w_2} = 2A_1/\pi$.

When the minimum value of w_2 is large compared to the range of the desirable output frequency, $w_1 + w_2$ or $|w_2 - w_1|$, the filter at the output of the mixer can suppress the unwanted spurs effectively.

Imperfections in the switch(es) introduce more undesirable IMD products. For example in the classical (balanced) diode mixer [4]-[5], the diodes biased by one of the signals act a the switch for the other one. A typical behavior of a doubly-balanced mixer is shown in Table II, where m, r refer to the IMD products at frequency $|m\, w_1 \pm r\, w_2|$.

Finally, the frequency mixing can be done semi-digitally, using for example an XOR gate as shown in Figure 7. In this case we have that

$$\hat{x}_1(t) = \frac{4}{\pi} \sum_{m=0}^{\infty} \frac{1}{2m+1} \sin\big((2m+1)\, w_1\, t\big)$$

and $\hat{x}_2(t)$ is given by (13). Multiplication of the two series results in the desirable signal component at frequency $w_1 + w_2$,

[6]For notational simplicity we assume the behavior of the switch does not alter the phase or amplitude of the output frequency components. Also, a gain factor may multiply expression (14).

[7]For simplicity we ignore the case where some frequencies (15) coincide.

		r :								
		0	1	2	3	4	5	6	7	8
m:	0	-	60	60	70	72	72	62	70	70
	1	60	0	35	15	37	37	45	40	50
	2	70	72	72	70	82	62	75	75	100
	3	75	63	66	72	72	58	86	58	80
	4	90	84	97	86	97	90	100	90	92
	5	90	84	86	72	92	70	95	70	92
	6	100	92	97	95	100	100	95	100	100
	7	100	97	102	95	100	100	100	90	100
	8	100	100	100	100	100	100	100	100	100

TABLE II

TYPICAL BEHAVIOR OF DOUBLY-BALANCED DIODE-MIXER. m, r REFER TO THE IMD PRODUCTS AT FREQUENCY $|m\,w_1 \pm r\,w_2|$. (FROM SYNERGY MICROWAVES CORPORATION'S APPLICATION NOTES [6]).

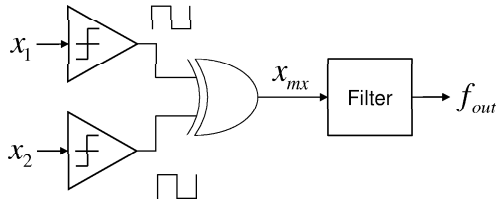

Fig. 7. Mixing of two square signals

or $|w_1 - w_2|$ with amplitude $A_{w_1 \pm w_2} = 8/\pi^2$ and the spurious components at frequencies

$$w^{\pm}_{(m,r)} = \left|(2m+1)w_1 \pm (2r+1)w_2\right| \tag{17}$$

with amplitudes[8]

$$A_{(m,r)} = \frac{A_{w_1 \pm w_2}}{(2m+1)(2r+1)} \tag{18}$$

where $m, r = 0, 1, 2, \ldots$ and $A_{w_1 \pm w_2} = 8/\pi^2$.

In general the XOR mixer introduces a lot of high level spurious signals. However, it can be a useful choice when $|f_1 - f_2|/|f_1 + f_2| << 1$ or when $f_1/f_2 << 1$, [7]. In particular, it can be used in the frequency offsetting technique below.

B. The Frequency Offsetting

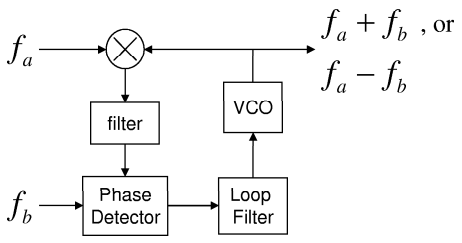

Fig. 8. Frequency Offset Loop: It provides either the sum $f_a + f_b$ or the difference, $f_a - f_b$ of the two input frequencies. It is a preferable mixing approach when $f_b << f_a$.

The frequency offsetting, or offset PLL concept is shown in Figure 8. It is an effective way for adding/subtracting a small frequency f_b to/from a large one f_a.

[8]For simplicity we ignore the case where some frequencies (17) coincide.

It can be used as a convenient alternative to mixing the signals when $f_{out} = f_a + f_b$ (or $f_{out} = f_a - f_b$) is very close to f_a, because the PLL's loop-bandwidth acts as a narrow band-pass filter centered at f_{out}. Instead, mixing f_a with f_b directly generates output spurious signals that require a very narrow-band filter to remove, which is usually challenging to build.

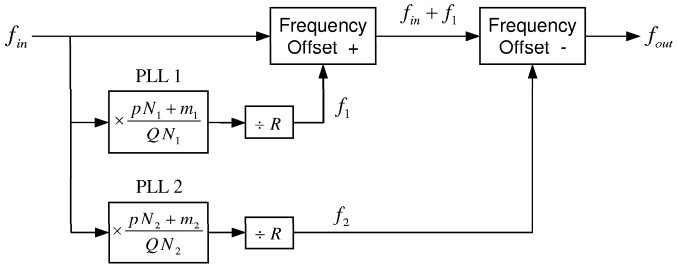

Fig. 9. A 2-PLL DFS scheme for timing applications using frequency offsetting [8].

Figure 9 shows a example of a DFS scheme, with two offset PLLs, for timing applications requiring very small output frequency range but very high resolution and signal purity [8].

V. CONCLUSIONS

Two important aspects of the application of Diophantine Frequency Synthesis (DFS) method have been discussed: The choice of the DFS parameters that lead to decimal output frequency resolution (10^mHz) for "standard" reference frequencies, and, the choice of the frequency mixer required in PLL-based DFS.

REFERENCES

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

[2] Paul P. Sotiriadis, "Cascaded Diophantine Frequency Synthesis, *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, Vol. 55, No. 3, Apr. 2008, pp.741-751.

[3] B. Gilbert, "A precise four-quadrant multiplier with subnanosecond response, *IEEE Journal of Solid-State Circuits*, Vol. 3, No. 4, Dec. 1968, pp. 365-373.

[4] Stephen A. Maas "*Microwave Mixers*", second edition, Artech House Publishers, 1993.

[5] Ulrich L. Rohde, "*Microwave and Wireless Synthesizers: Theory and Design*", first edition, Wiley-Interscience, 1997.

[6] Synergy Microwaves Corporation, Application Note / Tutorial on Mixers $http://www.synergymwave.com/$

[7] P. Sotiriadis, G. Weaver, "A Diophantine Frequency Synthesizer for the Examination of High Spectral Purity", *IEEE International Frequency Control Symposium 2007*.

[8] P. Sotiriadis, G. Weaver, "Diophantine Frequency Synthesizer Design for Timekeeping Systems", *International Journal of Navigation and Observation*, Vol. 2008, Article ID 416958 - *Invited Paper*.