Phase-Domain Fractional-N Frequency Synthesizers

Ayman M. ElSayed, *Member, IEEE,* and Mohamed I. Elmasry, *Fellow, IEEE*

***Abstract*—The concept of phase-domain fractional-N frequency synthesis is presented. Synthesizers using this architecture can achieve fast frequency switching without limiting the minimum channel spacing. In this architecture, a numerical phase comparator is used in conjunction with weighting coefficients, as a linear *weighted phase-frequency detector*. The synthesizer output spur level is determined by two factors. Namely, the delay of the numerical phase comparator, and the accuracy of the digital-to-analog convertor (DAC) used to convert the phase error to the analog domain. A novel second-order timing-error cancelation scheme is proposed to eliminate the effect of the phase comparator delays. Using this technique together with a 10-bit accuracy DAC, a maximum spur level of less than −65 dBc is simulated for a 900-MHz synthesizer. The settling time of the simulated synthesizer is less than 7 μs, and is independent of the channel spacing. The details of the synthesizer architecture, design considerations, and system-level simulations are presented. Implementation issues including the DAC accuracy and timing-error effects are discussed extensively throughout the text.**

***Index Terms*—Digital phase comparator, digital-to-analog converter (DAC), frequency synthesiszer, phase-frequency detector (PFD), phase-locked loop (PLL).**

I. Introduction

ONE OF THE important objectives of using fractional-N synthesizers is to relax the tradeoff between the frequency switching speed and the minimum synthesizer channel spacing. Generally, fractional-N synthesizers that use analog- or delay-based compensation provide a limited fractionality between modulo-8 and modulo-16. Often though, this is not enough to open up the loop bandwidth to achieve fast switching. Sigma–Delta fractional synthesizers [1]–[3] provide an unlimited resolution (depending only on the modulator number of bits), which allows high reference frequencies and a wider bandwidth. Unfortunately, the bandwidth increase is limited by the quantization noise of the Sigma–Delta modulator. Higher order Sigma–Delta modulators help reduce the low-frequency quantization noise at the expense of a fast increase of quantization noise with frequency. Therefore, the loop bandwidth has to be limited to suppress this noise. All digital direct digital frequency synthesizer (DDFS) solutions can provide much faster switching, and high resolutions but they do have a limited output frequency.

Manuscript received May 15, 2000; revised January 28, 2002. This work was supported in part by MICRONET and CITO and in part by Conxant Systems, Newport Beach, CA. This paper was recommended by Associate Editor U. Moon.

A. M. Elsayed was with the University of Waterloo, Waterloo, ON N2L 3G1, Canada. He is now with Intel Corporation, Sacramento, CA 95827 USA (email: ayman.m.elsayed@intel.com).

M. I. Elmasry is with the VLSI Research Group, University of Waterloo, ON N2L 3G1, Canada (email: elmasry@vlsi.uwaterloo.ca).

Digital Object Identifier 10.1109/TCSI.2003.820241

In this paper, we investigate a fractional-N frequency-synthesis technique that is based on using a numerical phase comparator to predict the phase error of the phase-locked loop (PLL). Numerical phase comparators were used in the past for several applications, and achieved promising results [4]–[7]. In this paper, an attempt is made to adapt the use of this technique for wireless applications.

In Section II, the concept of weighted phase error is presented, in which a numerical phase comparator is required to achieve channel selection. Then, a full-frequency synthesizer architecture based on this concept is presented in Section III. In this section, both the analog and digital implementation alternatives for this architecture are evaluated. The analog solution is simpler, but causes some uncertainty in the output channel frequency. This frequency error is not acceptable in most wireless systems; hence, the digital solution is favored. A novel timing-error correction scheme that renders the digital implementation more practical is also presented. Practical circuit design issues are discussed in Section IV. System level simulations setup and results are presented in Section V.

II. Concept of Weighted Phase Error and Channel Selection

Most fractional-N frequency synthesizers use a dual modulus divider in the feedback path of a PLL in order to achieve fractional division. Typically, this dual modulus divider utilizes a pulse swallow circuit to switch the division ratio between N and $N+1$ according to an input control signal. This control signal is alternated between zero and one in a controlled pattern such that the effective division ratio is N plus a controllable fraction f, which is determined by the switching pattern of the control signal. In this paper, this type of fractional-N synthesis is referred to as "*time-domain fractional synthesis*" (TDFS). Included in this category are analog compensated, delay compensated, and Sigma–Delta fractional-N synthesizers. They all rely on changing the division ratio in the time domain between multiple integer values in order to achieve an average fractional division, regardless of the spur compensation technique used.

In this paper, a technique for achieving fractional-N synthesis that has received very little attention in the literature, is evaluated. This technique is referred to as "*phase-domain fractional synthesis,*" and is explained next. The signal is instantaneously divided by multiple integers. The phase of each divider output is compared to the reference phase to produce a branch phase error. The total phase error can then be generated as a linear combination of these individual phase errors to produce an effective phase error, corresponding to a fractional divider. The fraction of the division ratio is determined by the ratio of the weighting factors used in the linear combination.

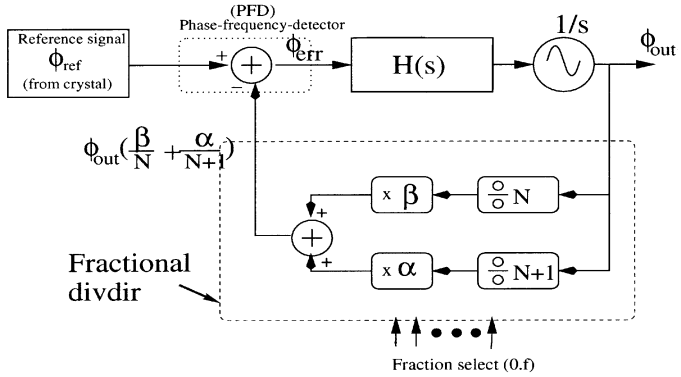


Fig. 1. Phase-domain frequency synthesizer (PDFS).

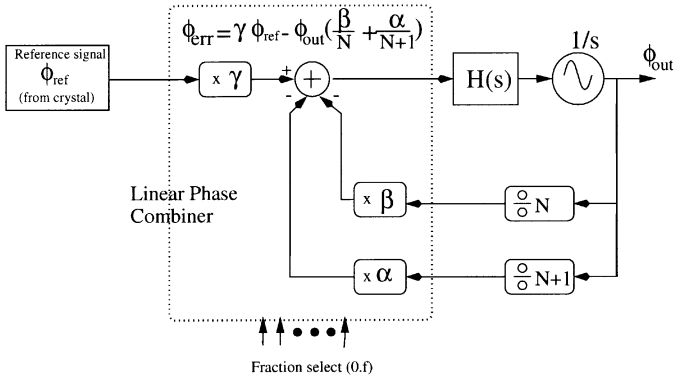


Fig. 2. Use of linear phase combiner for the PDFS.

In this scenario, a question may arise about how to monitor the individual phase errors. Due to the difference in the input frequencies, the phase-error values will grow indefinitely overflowing the system. Fig. 1 illustrates one possible structure of a frequency-domain fractional divider to overcome this overflow. Two integer frequency dividers are utilized. The first continuously divides by N, while the second always divides by $N+1$. The phases of the output signals from both dividers are linearly combined using the weighting factors α and β before they are compared to the reference. The output phase corresponds to a fractional division ratio $(N+f)$ between N and $N+1$. When the loop is locked the output phase error is always finite and the fractional division ratio is expressed as

$$N+f=\left(\frac{\beta}{N}+\frac{\alpha}{N+1}\right)^{-1}. \tag{1}$$

In order to provide a unique mapping between α, β, and f, one more constraint is required. An obvious choice that guarantees f to be bounded between zero and one, and to cover the full range $0 <= f \leq 1$, is to set $\alpha+\beta=1$ where $0 \leq \alpha, \beta \leq 1$. Using this constraint we get $N+f|_{\alpha=0,\beta=1}=N$ and $N+f|_{\alpha=1,\beta=0}=N+1$.

However, this constraint requires α and β to have infinite number of bit representation to provide equal channel spacing. To overcome this restriction the architecture of Fig. 1 is further modified to that of Fig. 2, where we add one more degree of freedom by multiplying the phase of the reference source by γ. The total phase-error output of the linear phase combiner is then expressed as

$$\phi_{\rm err}=\gamma\phi_{\rm ref}-\phi_{\rm out}\left(\frac{\beta}{N}+\frac{\alpha}{N+1}\right). \tag{2}$$

Most practical PLL synthesizers utilize a pure integrator in the loop filter [8] to null the phase error when the loop is locked. Setting the phase error of (2) to zero and choosing $\gamma=\alpha+\beta$, we can write the relation between the reference phase and the output phase as

$$\frac{\phi_{\rm ref}}{\phi_{\rm out}}=\frac{1}{N+f}=\frac{\frac{\alpha}{(\alpha+\beta)}}{N}+\frac{\frac{\beta}{(\alpha+\beta)}}{N+1}. \tag{3}$$

This relation is equivalent to that of (1). The difference is that α and β can now have any positive values. In order to limit the number of bits required to represent α and β we choose

$$\alpha+\beta=N+f \tag{4}$$

which results in

$$\alpha=f(N+1) \tag{4a}$$

$$\beta=N(1-f). \tag{4b}$$

From this result, it is obvious that the maximum number of bits needed to represent any of α, β, or γ is equal to the sum of the number of bits representing both the integer divisor (N) and its fraction (f). In other words, the required number of bits is determined by the frequency resolution, or the ratio of the synthesizer output frequency to the minimum channel spacing. For example, in a global system of mobile communications (GSM) phone, this ratio is 900 MHz/200 kHz which is equivalent to 13 bits.

The linearity of the "phase combiner" in Fig. 2 is critical to prevent mixing of input-frequency components causing fractional spurs at the VCO control input. If the combiner is perfectly linear, the frequency components at the output are only those fed at the input. Because this is a fractional division, the reference frequency can be chosen to be arbitrarily high. Therefore, a relatively wide-band low-pass filter in a fast switching PLL could be used to suppress these components. Practical phase combiners have different factors that limit their linearity and cause spurs at the synthesizer output. This limitation as well as the practical implementation of the synthesizer are discussed in the rest of this paper.

III. Proposed Fractional-N PLL Architecture

Both PLL architectures of Figs. 1 and 2, require two high-speed frequency dividers, as opposed to only one in conventional PLLs. This necessitates more power consumption as well as a complicated phase combiner design, which has to combine the phase of three signals. In order to reduce this extra hardware, we notice that the fractional divider of Fig. 1 has enough degrees of freedom to perform the fractional division, and the extra degree of freedom introduced by adding γ in Fig. 2 is only needed to limit the required number of bits for α and β representation. Therefore, we can remove one degree of freedom from

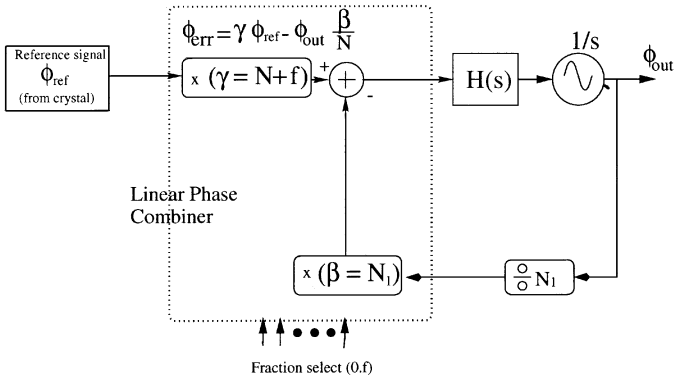


Fig. 3. Reduced complexity PDFS using one frequency divider.

the architecture of Fig. 2 and still obtain the required fractional division. Consequently, in Fig. 3, we remove the $(N+1)$ branch and attempt to obtain the fractional division with a finite bit representation for the two remaining coefficients γ and β. In order to avoid confusion, the integer divider value is referred to as N_1 which could have a different value than N; the integer part of the required frequency multiplier $N + f$. Setting $\alpha = 0$, and choosing

$$\beta = N_1 \quad \text{and} \quad \gamma = N + f$$

in (3), the effective division ratio reduces to $N + f$. The choice of N_1 is arbitrary; it does not have to be the same as N, as long as $\beta = N_1$. The proper choice of N_1 may reduce the hardware and/or the output spurs. In this PLL, not only is one divider branch and its associated hardware eliminated, but also the weighting factors are simplified to a direct mapping of N_1 and $N + f$.

Again, the key element in this architecture is the linear phase combiner. Apparently, we cannot use a conventional phase-frequency detector (PFD), because it has a limited range for linear operation that requires the two input frequencies to be equal when the loop is locked, which is not the case here. A linear phase detector that takes the frequency into account can be implemented as a counter (or an accumulator) that is incremented at each positive (or negative) edge of its input. Instead of two accumulators (one for each input signal) and a subtracter to detect the phase difference, one accumulator can be used to add γ at the reference edge, and to subtract β at the divider output edge. The accumulator content at any time becomes the weighted phase error we require. If the loop filter has a pure integrator pole, the average phase error is zero when the loop is in lock. Thus, the accumulator does not overflow as long as it has enough bits to represent its value variation around zero (two bits more than $N + f$ should be sufficient).

The output of the linear phase combiner should be in analog form to control the voltage-controlled oscillator (VCO) input. This leaves us with two choices. The first is to convert γ and β to analog signals and implement the phase accumulator in the analog domain. The second option is to build the accumulator in the digital domain and then convert its output to analog signal to drive the loop filter. Both techniques have their advantages and drawbacks and will be evaluated next.

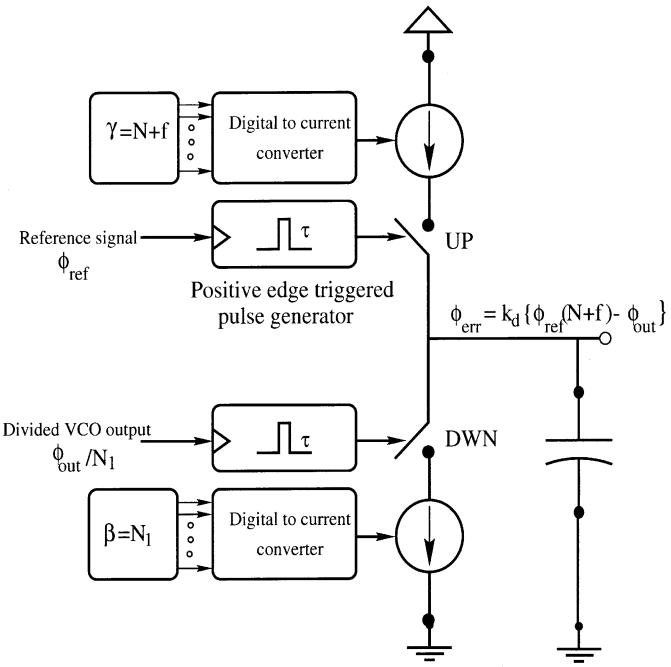


Fig. 4. Charge-pump-based analog linear weighted PFD (WPFD).

A. Analog PFD Implementation

The charge pump that is used in most PLL designs can be utilized to implement an analog accumulator. Fig. 4 describes the architecture of such an accumulator. The digital data words γ and β are used to program two digitally controlled charge pumps: one for the up current, and one for the down current. A digitally controlled charge pump, which we call a "*digital-to-current*" converter, is in fact a digital-to-analog converter (DAC), whose output is current rather than a voltage. The output current is integrated into an output capacitor to produce the phase error. The UP DAC requires a number of bits equal to that of $N + f$, whereas the DWN DAC needs only a number of bits equal to that of N_1. The integral nonlinearity (INL) requirements of those DACs, as well as the matching of their reference currents, are very strict. To explain this, the full operation of the linear phase combiner of Fig. 4 is described.

The positive edge pulse generator produces a pulse of a fixed duration (τ) at each positive edge of its input signal. The width of the pulses generated by both the reference signal and the divided VCO output should be well matched. At the positive edge of the reference, a charge equal to $(\tau.\gamma)$ is pumped to the output capacitor. At the divided VCO output positive edge, a charge equal to $(\tau.\beta)$ is drawn from the capacitor. The net capacitor charge is then proportional to the accumulated weighted phase of the reference (modified by its own DAC and pulse width error factor) minus that of the divided VCO output. An active loop filter is necessary to integrate the charge pump output which represents the phase error. In steady state, when the PLL is locked, the charge-pump capacitor voltage should be zero corresponding to a zero phase error. Thus, a differential charge pump is necessary to avoid saturating the DWN current source. Because of the fact that each phase is modified by its own error factor before the subtraction, a false zero phase error can be detected, leading to an error in the resulting synthesizer output

frequency. Although this error may be acceptable in some applications, it is not acceptable in cellular systems. For example, the required channel-frequency accuracy for a GSM systems is ± 50 Hz of the output frequency which is around 900 MHz. This translates to a 24-bit INL accuracy for the current DAC. Because the output capacitor integrates the current over time, all charge errors caused by charge injection, clock feedthrough, and finite output conductance in current sources adds to the output-frequency error. The same level of accuracy is also required for the ratio between the two DAC outputs. These numbers are impractical and limit this architecture to those applications with a large tolerance on the frequency accuracy.

Before we proceed to the digital alternative, we mention two advantages of this implementation. First, the current sources are independently controlled. This removes any restrictions on the time spacing between the edges of the input signals. Both signals could arrive at exactly the same with no restriction. The second advantage is that the DAC speed is not critical. It is determined by the required channel switching speed, which is in the microsecond or even the millisecond range. This allows the use of sophisticated techniques to implement those DACs such as digital Sigma–Delta modulators.

B. Digital PFD Implementation

In order to mitigate the frequency error inherent in the above analog accumulator based phase detector, a digital accumulator implementation is considered. Fig. 5 describes a frequency synthesizer based on a digital WPFD. The heart of this synthesizer is the accumulator register which holds the latest phase error value. A digital word γ is added at the positive edge of the reference signal. Another digital word β is subtracted at the positive edge of the divided VCO output. When the loop is in lock, the average accumulator contents, which represent the absolute phase of the reference multiplied by γ minus the absolute phase of the divided VCO output multiplied by β, is zero. After the signals are subtracted digitally, a DAC is used to convert the output to the analog domain before applying it to the loop filter. The DAC nonlinearity introduces similar errors to the analog WPFD case. The main difference is that the digital contents of the accumulator are always correct. The errors are introduced after the weighted phases are subtracted. As a result, the phase error does not accumulate, and the error in determining the center frequency becomes zero. Once the frequency is locked correctly, any inaccuracy in sensing the phase error due to the DAC (including quantization error, nonlinearly and/or dc offset), is transformed into spurs and does not affect the frequency accuracy. This might seam counter intuitive in the case of dc offset and is explained by comparing the continuous time equivalent PFD phase error output in this digital implementation to that of the previous analog one

$$\phi(t)_{\text{err analog WPFD}} = \int \left[(A_1\gamma + k_1) f_{\text{ref}} - \frac{(A_2\beta + k_2) f_o}{N_1}\right] dt \tag{5}$$

$$\phi(t)_{\text{err digital WPFD}} = A_3 \int \left(\gamma f_{\text{ref}} - \frac{\beta f_o}{N_1}\right) dt + k_3 \tag{6}$$

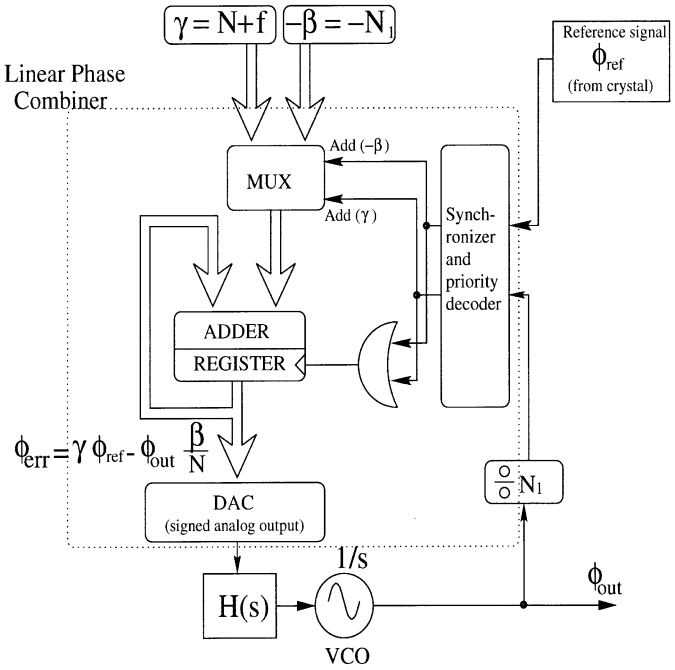


Fig. 5. Full PLL frequency synthesizer using digital accumulator as a WPFD.

where A_i and k_i represent reference scaling and the offset of the ith DAC. Unlike the analog accumulator case, forcing the digital WPFD output to zero (through type-2 PLL) leads to

$$\int \left(\gamma f_{\text{ref}} - \frac{\beta f_o}{N_1}\right) dt = -\frac{k_3}{A_3} \quad \text{or}$$

$$f_o = \frac{\gamma N_1}{\beta} f_{\text{ref}}. \tag{7}$$

This output-frequency expression is exact and is independent of the DAC error or dc offset. The dc offset will only cause the loop to lock to a nonzero phase shift between the PFD inputs causing periodic spurs at the output. This is not the case with the analog accumulator solution because the offset in that case is translated to an error in the values of γ and β causing the loop to lock to the wrong frequency.

The output spurs are caused by charge error due to not only the DAC imperfection, but also the timing delay of the accumulator. The effect of this timing delay is described in the first three waveforms in Fig. 6. Waveform (c) shows a sample of the accumulator contents in response to an input reference (a), and a divided VCO output (b). In ideal conditions when the loop is in lock, the average area under trace (c), which corresponds to the integrated phase error, should be equal to zero. An error in either the amplitude (due to the DAC) or the timing (due to finite adder delay) causes the loop to detect a false shift from the zero average, and attempts to compensate for it. Because the error does not accumulate, the loop soon detects the erroneous phase shift it has caused, and corrects for it again. The same scenario is repeated in a periodic manner, causing the output spurs. Several factors determine the period of the spurs, and they are best predicted by simulation. Among these factors are the DAC number of bits, the DAC accuracy, the desired output frequency,

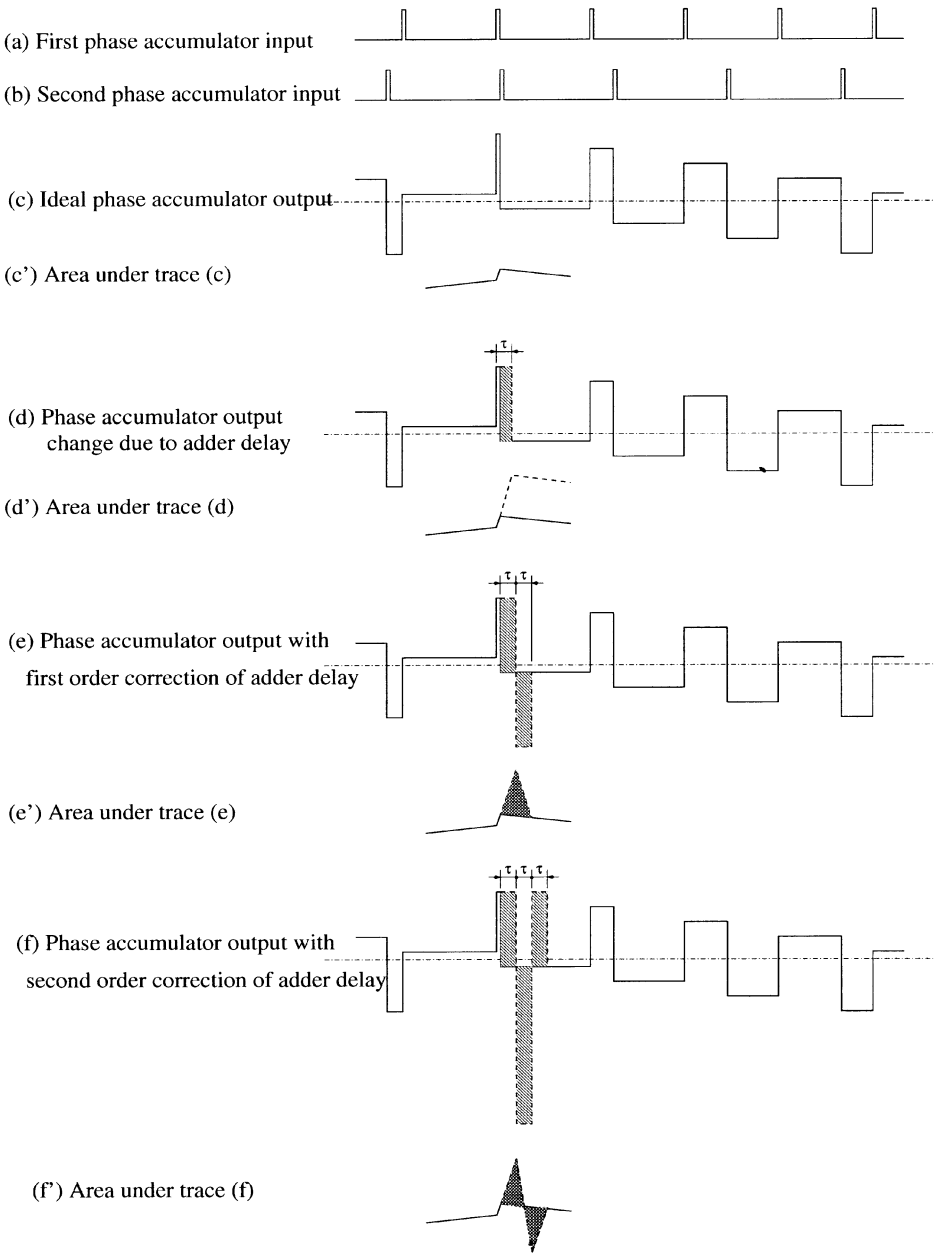


Fig. 6. WPFD adder delay effects and correction schemes.

and the factors γ and β. Proper frequency planning through a good choice of N_1 and β can help move the spurs away from the critical blocking channel frequencies. The spur levels are directly proportional to the errors introduced and are managed only by controlling these errors, both in timing and amplitude.

1) Timing Error: In the following discussion, the timing error is shown to be more serious than the amplitude one. The error caused due to the adder delay is depicted in Fig. 6. Both the reference signal and the divided VCO outputs run at different frequencies, and are not synchronized. Consequently, the positive edges of both signals could occur within a very short (or even zero) time interval. Because the accumulator cannot change its state instantaneously, it either ignores the second incoming pulse causing a huge unrecoverable phase and frequency error, or it responds to it after a certain delay. In this second case, no frequency error occurs and only a spur is caused at the output, similar to the effect of the DAC errors. Trace (d) in Fig. 6 compares this second case with the ideal situation of (c). The shaded area in trace (d) represents the error introduced by the delayed accumulator response to one of its two inputs. The resulting phase integrator outputs are shown in traces (c') and (d'). Because the delay error represents a relatively large fraction of the input signal periods, the error it introduces is equivalent to the use of a very limited number of bits of the DAC. For example, assume the reference frequency is 50 MHz (20-ns period), and the accumulator DAC delay is 3 ns. The error introduced by this delay corresponds to the use of a 3 bit DAC.

In order to reduce the effect of this delay error, the accumulator operation is modified to self correct the resulting error in the VCO control signal. Trace (e) of Fig. 6 shows the phase error output of a proposed solution. The resulting integrated

phase output is shown in trace (e'). If signal (b) arrives while the accumulator is busy processing signal (a), signal (b) is passed through two identical unit delay blocks. The duration of each of these unit delays is longer than the accumulator processing time. After passing through the first unit delay, signal (b) prompts the accumulator to subtract a value of 2β from its contents. After the second unit delay, the value of β is added. The resulting integrator output at the end of these two delays, as shown in trace (e'), is identical to that of the ideal case of trace (c') which suggest full removal of the delay error. Although delay caused spurs are significantly reduced using this scheme, system simulations still show spur levels as high as -40 dBc when the accumulator DAC delay is not set to zero.

Further investigation of the loop dynamics reveals that the subsequent filtering stages, after the phase integrator stage, perform another averaging of the integrator output. This averaging is equivalent to a second integration. Hence, in order to completely cancel the delay error effect, the area under trace (e') should be the same as that under trace (c') which is not the case in this architecture. Therefore, a second-order correction scheme is proposed. As explained in trace (f), signal (b) is passed through three identical unit delay blocks. After the first delay the value 3β is subtracted from the accumulator. After the second delay, the value 3β is added. Finally, a value β is subtracted after the third delay. The resulting integrator output is shown in trace (f'), which has the same area under it as that of trace (c'). Simulations demonstrate a full cancelation of spurs caused by the delay error using this second-order correction scheme. Perfect matching of the unit delay blocks is crucial in achieving this result and is addressed in Section IV-B.

2) DAC Accuracy: With the second-order delay compensation in place, DAC imperfections rise again as the main source of the remaining spurs. There is no straightforward relation between the DAC number of bits, and the synthesizer output spurs. That is because the process is involved and both the level and the frequency of the spur depend on the number of bits, as well as the loop filter and divider parameters. Simulation results presented in Section V, show that an 8- to 10-bit DAC suffices for most applications. The spur levels do not increase if we increase the accumulator size without changing the DAC number of bits. As a result, the frequency resolution can be arbitrarily increased at the power and area expense of adding extra bits to the adder and the register only. In addition to the quantization noise, rise- and fall-time mismatches and switching glitches of the DAC also contribute the charge error responsible for the spurs. In order to minimize this effect, the output is allowed to settle for a relatively long time (~ 3.0 ns) in each state before the next input pulse.

IV. Circuit Design Issues

Although a full circuit implementation of the PDFS architecture is beyond the scope of this work, the objective of this section is to highlight some of the practical issues involved in such design.

A. Digital Delay and Power Consumption

The design of the digital WPFD, its relevant controls, and the digital decoder for the DAC, necessitates a fast, yet power efficient logic style. The accumulator which uses about 16-bit adder requires a delay in the range of 1 to 2 ns. While it is not easy to achieve such high speeds using conventional CMOS implementations, current-mode logic (CML) consumes too much power. Most of the CML power consumption results from static current, which is justified only if the logic runs at very high clock rates. Although it needs high processing speed, the WPFD runs at rates around the reference frequency (≤ 50 MHz), which makes the CML inefficient. An alternative logic style that has delays comparable to CML but only uses dc current when an evaluations is needed may be DyCML [9] or Sc^{2}L [10]. These logic styles meet our delay requirement, and their power consumption corresponds to the actual data rate rather than computation speed.

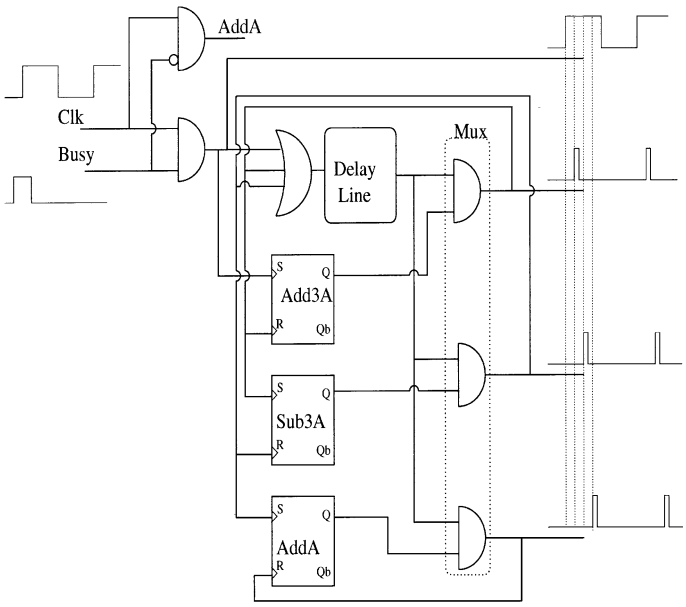


Fig. 7. Generating three equally delayed versions of "Clk" using a single delay line.

B. Delay Correction Accuracy

Another important issue that arises when designing the delay compensation circuitry is how to match the different delays needed to achieve full spur cancelation. In order to avoid the mismatches among various delay components, we propose using a single delay cell with a simple finite state machine to replace the three identical delays needed for the second-order timing-error cancelation. An example implementation using 3-SR flip-flops is shown in Fig. 7. If the positive edge of the input Clk arrive while the busy signal is asserted, Clk sets "Add3A" flop and then goes through the delay line once to generate the first delayed version of Clk. This first delayed Clk then resets "Add3A" flop and instantaneously sets "Sub3A" flop before going through the delay line to generate the second delayed Clk. Similarly, the third delayed Clk version is created and then used to reset the the third flop "AddA." The only source of error in this implementation is the delay mismatch of the "MUX" AND gates which is much smaller than the main delay line.

The error cancelation accuracy is also affected by the rise and fall times of the DAC outputs, which directly relate to those

of the digital WPFD. This effect could be significantly reduced by matching the rise and the fall times, which is best achieved using differential architectures for both the digital WPFD and the DAC. Another helpful technique is to increase the delay line value so that the rise and fall times represent a small fraction of this period. This delay value is limited by the faster of the reference signal or the divided VCO output. In order to find the maximum delay τ, we assume that the period of the faster of these two signals is $T_{\min}$. In the worst case when the second signal arrives at the end of the busy interval τ, due to a previous signal, another 3τ delay is required for error cancelation plus one more τ to clear the accumulator for the next incoming signal. Therefore, when choosing τ we have to guarantee that

$$\tau \leq \frac{T_{\min}}{5}. \tag{8}$$

In the simulation example of Section V, $T_{\min}$ is approximately 17 ns, leading to $\tau \approx 3$ ns.

C. DAC Speed and Linearity

The most challenging circuit design in the digital WPFD architecture is the DAC. The second-order timing-error compensation greatly reduces the speed requirements of the DAC and brings it into a feasible range. For the above example, the DAC is required to resolve samples with a minimum interval of 3 ns. The required DAC accuracy is in the range of 8–10 bits depending on the required spur level. Many DAC designs that meet such requirements [11]–[13] have been reported. Most of those designs are current-mode designs that feed all the currents directly to the output. This is exactly what is needed for the WPFD DAC. The only problem with these designs is their relatively high power consumption for both the fast digital decoding of the inputs, and to achieve a high dynamic range at the output. The digital power consumption for the digital parts can be much lower in our case, due to the low average clock speed. In the WPFD DAC, the output current is integrated into the output capacitor, rather than being converted to a voltage through a resistor. Consequently, no voltage dynamic range problem exist, and small currents can be used.

In order to achieve the required accuracy with reasonable device matching, the design should be segmented into thermometer coded, and binary weighted parts [14]. The full binary weighted design is much simpler because no decoding is involved. However, it suffers from severe matching requirements and large transition glitches. A full thermometer coded design avoids these problems, but requires full decoding, and a large number of current cells. Because of area issues, segmentation is a good compromise, and is recommended for the WPFD DAC design.

V. System Simulations

In order to test the functionality of the WPFD-based synthesizer architecture, a Matlab Simulink system based on the block diagram of Fig. 5 is used. A simplified block diagram of the simulated system is shown in Fig. 8; e.g., the frequency bands shown correspond to a GSM receiver. The high input reference frequency of 51.2 MHz allows a wide-loop bandwidth, and fast

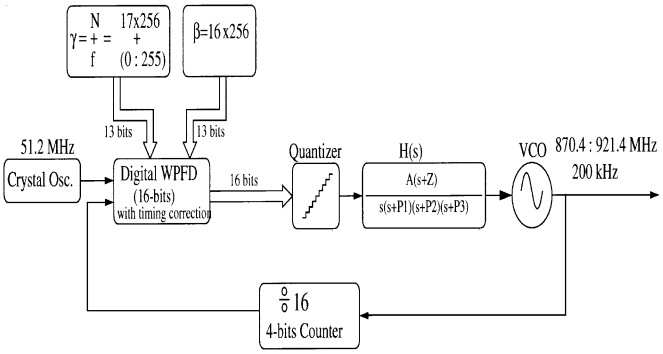


Fig. 8. Block diagram for WPFD simulations.

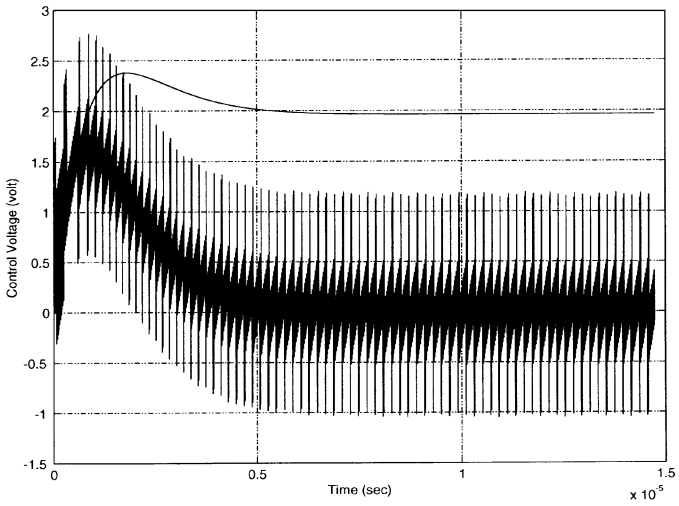


Fig. 9. Transient response of WPFD PLL.

switching times. A 13-bit register is used for γ, out of which 8 bits are used for the fraction leading to a channel resolution of 51.2 MHz/256 or 200 kHz. The GSM band is covered by changing the fraction value only; the integer multiplier is kept at 17 all the time. A 4-bit counter provides a fixed divide-by-16, with a corresponding β value of 16. The VCO gain is 50 MHz/V, the PLL natural frequency is 190 kHz, and the damping factor is 0.707. Two secondary poles at 1.6 and 5.4 MHz help reduce the reference spurs. The VCO control as well as the WPFD accumulator value (or the phase error) are plotted in Fig. 9, for a 100-MHz output frequency step. The settling time is approximately 7 μs. The thin phase error excursions in the figure are a result of the second-order timing correction.

The VCO output spectrum is shown in Fig. 10. For this plot, a 10-bit DAC is used with a second-order timing correction and a τ value of 3 ns. The integer part of γ is 17, and fraction is 137/256. The resulting output frequency is 898.2 MHz. The simulation step is 13.13 ps, and four Mega samples are used for the fast Fourier transform. The reference and the divided VCO spurs are at -92 and -96 dBc below the carrier, respectively. The close-in spurs due to the DAC quantization error are below -65 dBc.

In addition, the effects of both the first-order and the second-order delay correction schemes are examined. For this experiment, an ideal DAC is assumed for all cases. The WPFD DAC delay τ is set to 3 ns. Fig. 11 illustrates the output spectrum

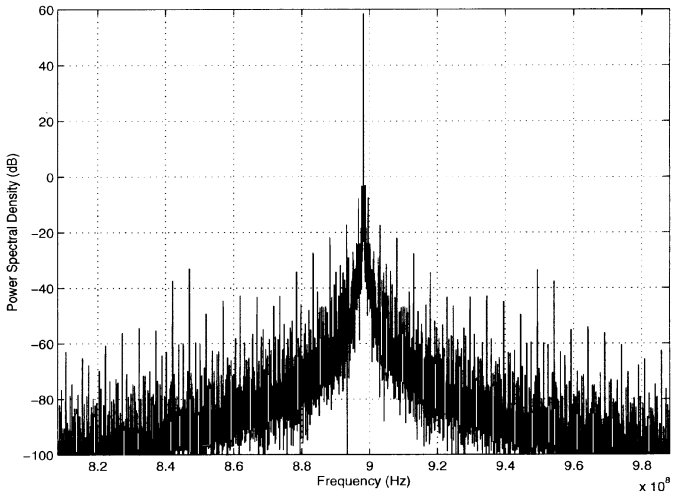


Fig. 10. PLL output spectrum for a 10-bit DAC, and second-order timing-error correction.

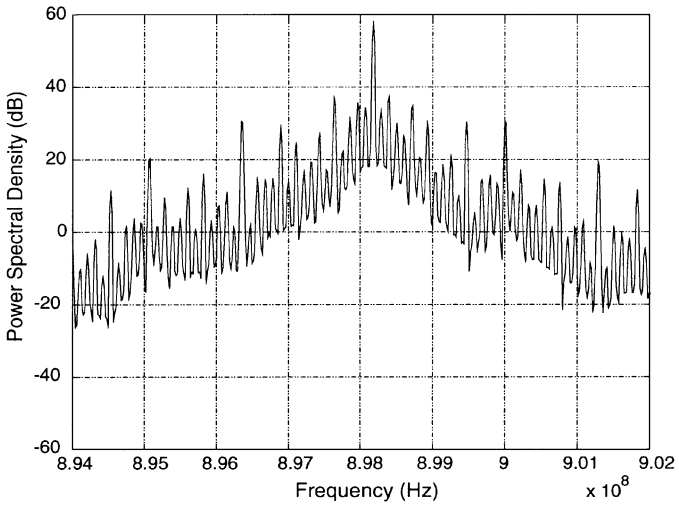


Fig. 11. PLL output spectrum for an ideal DAC, and no timing-error correction.

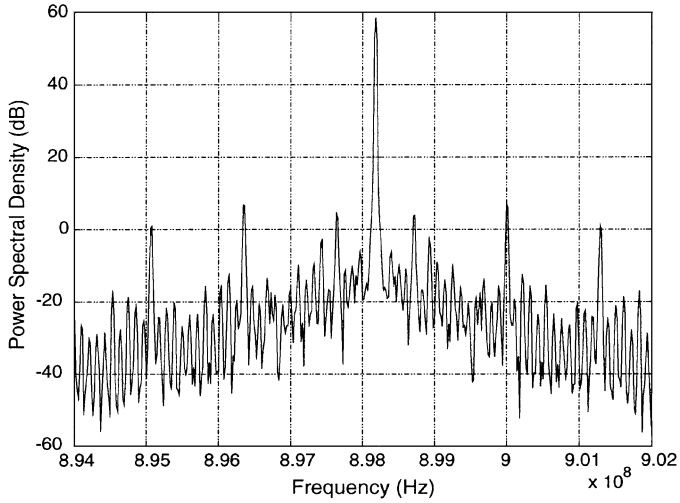


Fig. 12. PLL output spectrum for an ideal DAC, and a first-order timing-error correction.

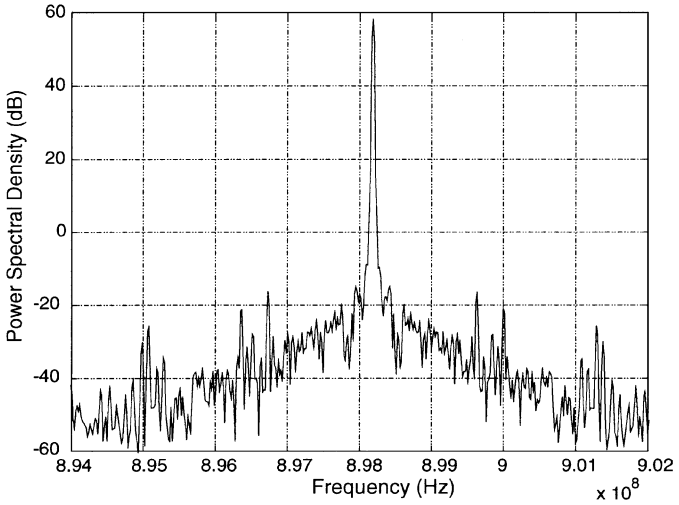


Fig. 13. PLL output spectrum for an ideal DAC, and a second-order timing-error correction.

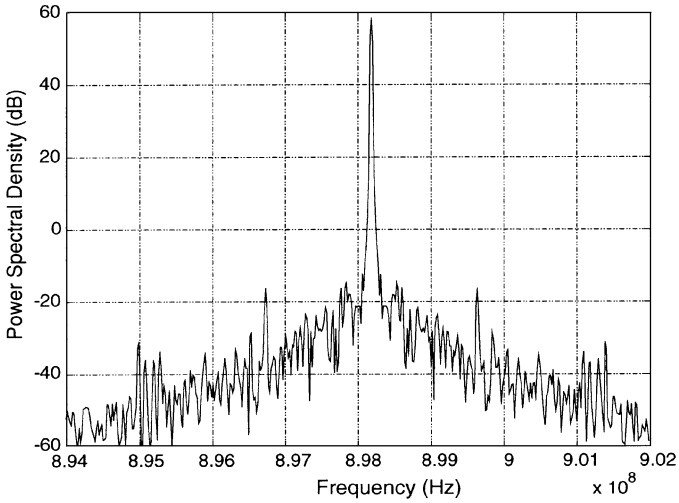


Fig. 14. PLL output spectrum for for ideal DAC, and a delay free accumulator.

when no delay correction is used. The close-in spurs are as high as -20 dBc. The use of the first-order delay correction drastically reduces the spurs as portrayed in Fig. 12. The spurs are at -51 dBc, which is still too high for many applications. Further reduction in the spur levels is achieved when a second-order correction is employed. The resulting spectrum is shown in Fig. 13, where the spurs are below -77 dBc. This spur level is equal to that resulting when an ideal accumulator (with no delays) is used (Fig. 14). Therefore, the second-order correction does provide almost a complete timing-error cancelation.

In order to determine the required DAC accuracy, we vary the quantizer number of bits to see its effect on the close-in spurs. The output spectrum using 8-, 9-, and 10-bit quantizers are illustrated in Figs. 15–17, respectively. The spur level decreases from -50 dBc when an 8-bit DAC is used, to below -65 dBc when a 10-bit DAC is used. This number is sufficient for most cellular applications. Consequently, a 10-bit DAC suffices for most complicated systems. A smaller number of bits can be used for less demanding systems.

In order to confirm that the use of quantization error is representative of the effective DAC accuracy required regardless

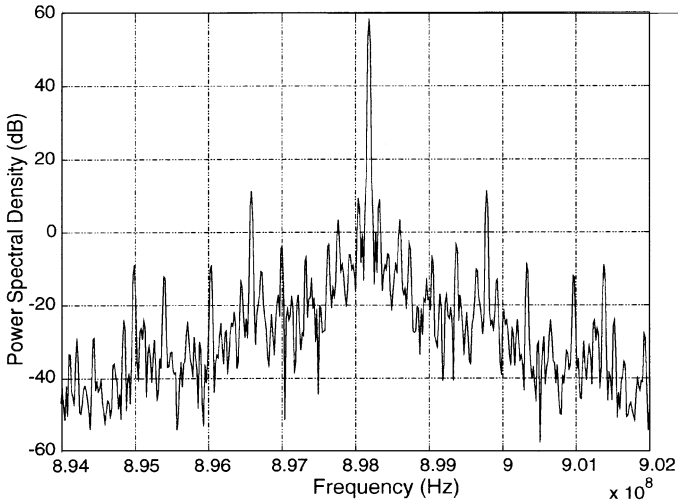


Fig. 15. PLL output spectrum for an 8-bit DAC, and a second-order timing-error correction.

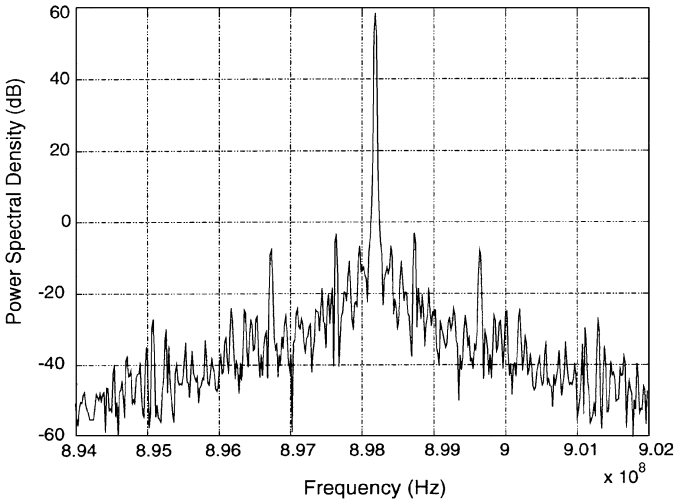


Fig. 17. PLL output spectrum for a 10-bit DAC, and a second-order timing-error correction.

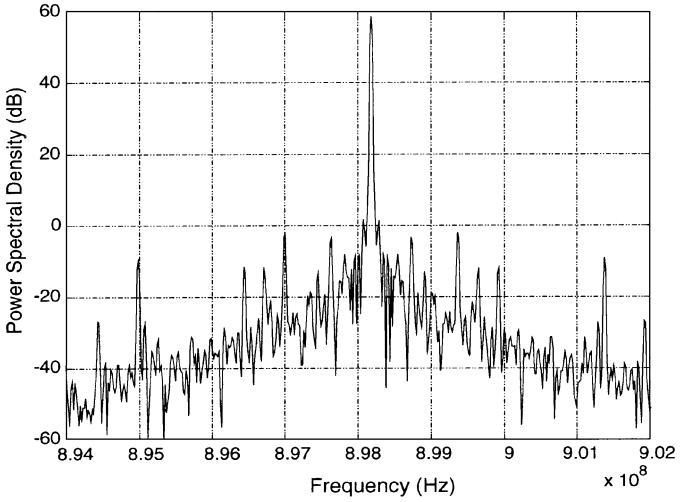


Fig. 16. PLL output spectrum for a 9-bit DAC, and a second-order timing-error correction.

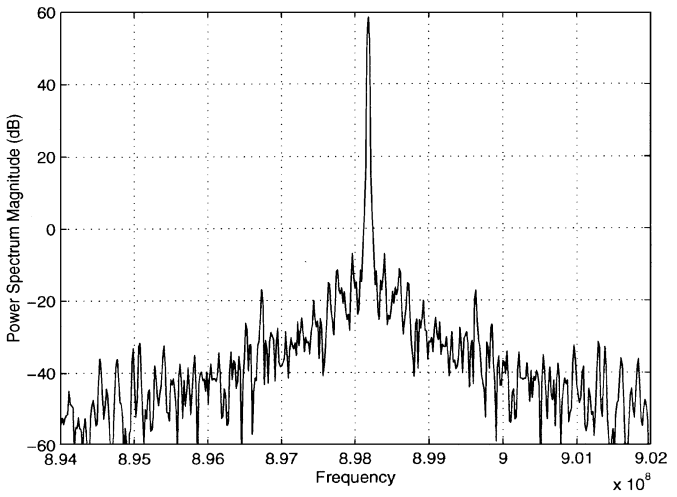


Fig. 18. Simulation results using a 13-bit quantizer with an 8.36 LSB INL.

to the source of error (quantization, INL or dc offset) we run a second 10-bit DAC accuracy experiment. Fig. 18 shows the simulation results using a 13-bit quantizer with an 8.36 LSB INL, which represents a 10-bit effective DAC. Spur levels are slightly lower than the case of the ideal 10-bit quantizer of Fig. 17. Therefore, it is safe to say that a 10-bit effective DAC guarantees spur levels to be below -65 dBc.

VI. Conclusion

A numerical phase comparator is used to construct a fractional-N PLL frequency synthesizer based on the weighted phase error concept. This architecture provides narrow channel spacing while allowing reference frequencies up to 50 MHz/s. Channel resolution is only limited by the WPFD accumulator size. The high-reference frequency greatly reduces the effective division ratio, thus, attenuating the in-band phase noise. In addition, the high-reference frequency allow wider loop bandwidth to increase the switching speed.

A novel timing-error cancelation technique is proposed to remove the output spurs due to the delays in the digital phase comparator and its subsequent DAC. The output spur level is limited only by the DAC accuracy, and the spur frequency is not related to the channel spacing. The frequency of the spur can be shifted by changing the frequency divider ratio N_1, such that the spur does not fall on the blocking channel. Spur levels as low as -65 dBc can be achieved if a 10-bit DAC is used. Unlike the $\Sigma - \Delta$ techniques the quantization noise does not increase at high frequencies. As a result, a much larger loop bandwidth is possible. The simulated switching time for a 900-MHz synthesizer with a 100-MHz frequency step is less than 7 μs.

Acknowledgment

The authors would like to thank A. Ali (Skyworks, Newport Beach, CA), A. Coban (Mindspeed, Newport Beach, CA), and A. Hafez (MemsCap, Cairo, Egypt), for very fruitful discussions.

REFERENCES

[1] T. A. Riley, M. Copland, and T. Kwasniewski, "Delta–Sigma modulation in fractional-N frequency synthesis," *IEEE J. Solid-State Circuits*, vol. 28, pp. 553–559, May 1993.

[2] N. M. Filiol, T. A. Riley, C. Plet, and M. Copland, "An agile ISM band frequency synthesizer with built-in GMSK data modulation," *IEEE J. Solid-State Circuits*, vol. 33, pp. 998–1008, July 1998.

[3] W. Rhee, A. Ali, and B.-S. Song, "A 1.1 GHz CMOS fractional-N frequency synthesizer with a 3b 3rd-order $\Sigma\Delta$ modulator," in *Dig. Tech. Papers ISSCC'00*, Feb 2000, pp. 198–199.

[4] R. J. Bosselaers, "Phase locked loop including an arithmetic unit," U.S. Patent 3 913 028, Oct. 1974.

[5] A. Kajiwara and M. Nakagawa, "A new PLL frequency synthesizer with high switching speed," *IEEE Trans. Vehi. Technol.*, vol. 41, pp. 407–413, Nov. 1992.

[6] M. Kokubo, K. Hori, T. Ito, Y. Tazaki, and N. Takei, "A fast-frequency-switching PLL synthesizer LSI with a numerical phase comparator," in *Dig. Tech. Papers ISSCC'95*, Feb. 1995, pp. 260–261.

[7] M. Zuta, "A new PLL with fast settling time and low phase noise," *Microwave J.*, pp. 94–108, June 1998.

[8] R. E. Best, *Phase-Locked Loops Theory Design and Applications*, 2nd ed. New York: McGraw Hill, 1992.

[9] M. Allam and M. Elmasry, "Dynamic current-mode logic (DyCML), a new low-power high-performance logic family," in *Proc. IEEE Custom Integrated Circuits Conf.*, May 2000, pp. 421–424.

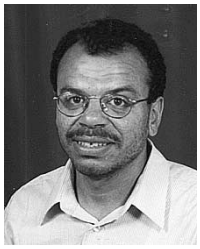
[10] A. M. Fahim and M. I. Elmasry, "Sc^{2}L; a low-power high performance dynamic differential logic family," in *Proc. Int. Symp. Low Power Electronics and Design*, Aug. 1999, pp. 88–90.

[11] A. Marques, J. Bastos, A. Van den Bosch, J. Vandenbussche, M. Steyaert, and W. Sansen, "A 12b accuracy 300 MSample/s update rate CMOS DAC," in *Dig. Tech. Papers ISSCC'98*, Feb. 1998, pp. 216–217.

[12] P. Vorenkamp, J. Verdaasdonk, R. van de Plassche, and D. Scheffer, "A 1 Gs/s, 10b digital-to-analog converter," in *Dig. Tech. Papers ISSCC'94*, Feb. 1994, pp. 52–53.

[13] C.-H. Lin and K. Bult, "A 10b 250 MSample/s CMOS DAC in 1 mm^2," in *Dig. Tech. Papers ISSCC'98*, Feb. 1998, pp. 214–215.

[14] R. van de Plassche, *Integrated Analog-to-Digital and Digital-to-Digital Converters*. Dordrecht, The Netherlands: Kluwer, 1994.



Ayman Elsayed (S'95–M'00) was born in Cairo, Egypt, in 1967. He received the B.S. and M.S. degrees in electrical engineering from Ain Shams University, Cairo, Egypt, in 1989, and 1994 respectively, and the Ph.D. degree in RFIC design from the University of Waterloo, Waterloo, ON, Canada, in 2000.

He was a Research Assistant in the VLSI Research Group, University of Waterloo, from 1994 to 1996. His M.S. research focused on the physics of multiple-quantum-well semiconductor lasers, and the focus of his Ph.D. dissertation was on the design of fully integrated low-phase-noise voltage control oscillators (VCOs) and fractional-N phase-locked loop (PLL) frequency synthesizers. During the course of his study, he held two internships in 1996 and 1998, with Conexant Systems, Newport Beach, CA, where he was invoved in the design of integrated VCOs and PLL frequency synthesizers. Since 2000, he has been with Intel Corporation, Sacramento, CA, where he is currently working as a Staff RFIC Design Engineer. His research interests are RF CMOS circuits, VCOs, frequency synthesizers, as well as system-level optimization of wireless receivers and transmitters. He holds one U.S. patent.

Mohamed I. Elmasry (S'69–M'73–SM'79–F'88) was born in Cairo, Egypt, in 1943. He received the B.Sc. degree from Cairo University, Cairo, Egypt, and the M.A.Sc. and Ph.D. degrees from the University of Ottawa, Ottawa, ON, Canada, in 1965, 1970, and 1974, respectively, all in electrical engineering.

He has worked in the area of digital integrated circuits-and-system design for the last 38 years. He was with Cairo University from 1965 to 1968, and with Bell-Northern Research, Ottawa, ON, Canada, from 1972 to 1974. He has been with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada, since 1974, where he is a Professor and Founding Director of the VLSI Research Group, also holding the University Research Fellowship, and was the NSERC/BNR Research Chair in VLSI design from 1986 to 1991. He has served as a Consultant to research laboratories in Canada, Japan, and the U.S. He has authored and coauthored over 400 papers and 16 books on integrated-circuit design and design automation. He has several patents to his credit. He is the founding President of Pico Electronics Inc. Waterloo, ON, Canada.

Dr. Elmasry has served in many professional organizations in different positions and received many Canadian and International Awards. He is a Founding Member of the Canadian Conference on VLSI, the Canadian Microelectronics Corporation, the International Conference on Microelectronics, the Canadian Research Centers of Excellence, MICRONET, and CITO. He is a Fellow of the Royal Society of Canada and of the Canadian Academy of Engineers.