A -107dBc, 10kHz Carrier Offset 2-GHz DLL-Based Frequency Synthesizer

Jingcheng Zhuang, Qingjin Du, Tad Kwasniewski

Department of Electronics, Carleton University
Ottawa, Ontario, Canada

Abstract
A low phase noise, low spur, DLL-based frequency synthesizer utilizing a novel charge pump phase comparator and edge combining circuit to reduce the jitter and spur is reported. To prove its feasibility, a 9-time multiplication frequency synthesizer is designed and fabricated in 0.18μm CMOS technology with an active area of 0.05mm². The output exhibits a spur power of -46.17dB below carrier and a phase noise of -107.17dBc/Hz at 10kHz offset with a reference frequency of 173MHz from a RF generator.

Introduction
The frequency synthesizer is an important building block in most communication systems, on whose performance the phase variance of frequency synthesizer output, including the random phase noise and spur-associated variance, has deleterious effects (1). Until recent times, synthesizer design was centered on the conventional PLL, but such devices suffer from the high phase noise output which is an inherent result of jitter or noise accumulation in voltage controlled oscillators. In recent years therefore, DLL-based frequency synthesizers which do not exhibit this accumulation over cycles of operation have been drawing more and more attention in this field. Indeed, some DLL-based frequency synthesizers have appeared in the literature (2,3) and their phase noise performance shows significant improvement over their PLL-based counterparts. However the reported DLL-based frequency synthesizers exhibit high levels of jitter and spur due to mismatches involving the edge combining circuit and in-lock errors. In this paper, the edge combining operation is analyzed in frequency domain, and a DLL-based frequency synthesizer which uses a novel charge pump phase comparator to achieve a great reduction of in-lock error, a highly-symmetric edge combining circuit to minimize the inter-period variance, and a lock detection circuit to guarantee correct locking is designed and fabricated. Actual measurement results show that this design results in improvements over contemporary methods of both phase noise and spur performance.

Operating Principle
Fig. 1 shows the block diagram of proposed DLL-based frequency synthesizer. The DLL itself consists of a voltage-controlled delay line (VCDL), a charge pump phase comparator and a loop filter. The VCDL input/output phase difference is smoothed by the low-pass loop filter to form a control voltage, this is then fed back to the VCDL to adjust its delay.

When the delay of the VCDL is locked to one period of the reference signal, an output signal whose frequency is a multiple of the input frequency is obtained by combining the delay stages' outputs of the VCDL.

To minimize the inter-period jitter which is in part due to the mismatch among the delay stages, the highly symmetric edge combining circuit shown Fig. 2 is used. Each delay stage outputs a pulse PU_n having a width of half its delay time. All of these PU_0, PU_1... PU_N-1 are sent to a pulse combiner which generates the output signal. With this structure, only the rising edges of the reference signal are used and consequently the frequency synthesizer output is immune to any duty-cycle asymmetry in the reference signal. Theoretically, if all delay stages have identical delay and their sum is exactly one period of the reference signal, the output whose frequency is N times of the reference frequency is spur free (given an absolutely symmetric edge combiner). However, in a real circuit, the above conditions cannot be satisfied exactly and some spurious frequencies show up in the frequency synthesizer output spectrum.

Assuming that all delay and pulse generation stages are identical (so that PU_0, PU_1... PU_N-1 have exactly identical shape...
except for some phase shift), the output, the sum (logic "OR") of all \( P_{U_0} \) to \( P_{U_{N-1}} \) can be found as follows.

Since the fundamental frequency of \( P_{U_n} \) equals to the reference frequency \( f_r \), one can write the Fourier transform of \( P_{U_n} \) as \( X_{P,n}(k,f) \), where \( n=0...N-1 \) and \( k \) is an arbitrary integer. If the delays between \( P_{U_n} \) and \( P_{U_0} \) are denoted \( t_d(n) \), then \( X_{P,n}(k,f) \) can be obtained as,

\[
X_{P,n}(k,f) = X_{P,0}(k,f)\exp(j2\pi f t_d(n)k)
\]

The Fourier transform of the output signal is the sum of these,

\[
X_{P,\text{out}}(k,f) = \sum_{n=0}^{N-1} X_{P,n}(k,f) \exp(j2\pi f t_d(n)k)
\]

Which shows that if,

\[
t_d(n) = \frac{n\pi}{N}
\]

where \( T \) is one period of the reference signal, \( X_{P,\text{out}}(k,f) \) is zero for all values of \( k \) except for \( k=mN \), where \( m \) is an arbitrary integer. So the output frequency is \( N \) times of the reference frequency and no spur appears. The condition of (3) should be satisfied as closely as possible in order that the output spur or jitter can be minimized. The accuracy of the Fourier sum of (2) depends on the condition of (3) as well as on the inter-stage delays being identical and the edge combining circuit being symmetrical.

**VCDL and Edge Combiner**

As shown in Fig. 3, each delay stage of VCDL consists of two inverting cells whose delays are controlled by two control signals \( V_{\text{con}}+ \) and \( V_{\text{con}}- \) coming from a drive circuit. Each delay cell consists of 3 inverters, and most delay of the delay cell is due to the first tunable inverter with the other two acting as an output buffer which isolates the delay cell and minimizes the effect of an unbalanced load. That two inverting delay cells are put together working as one delay stage may cancel the variation of duty cycle. Symmetry of the edge combining circuit is guaranteed by the use of a wire-OR gate to combine the pulses. In the actual design, two extra delay stages working as an input buffer are inserted at the beginning of the delay line and a single extra stage is placed at the end to balance the load of the last delay stage and provide a signal for the lock detector and window generator described below. Consequently, for a 9-time multiplication frequency synthesizer (\( N=9 \)), there are total 11 delay stages, and only 9 (\( n=0,1,...8 \)) of them are involved in delay locking and edge combining operations.

**Lock Detector & Window Generator**

Normally, phase detectors have a relatively narrow operating range and it can only sense the phase difference equivalent to less than half a cycle. To prevent the DLL from locking to zero cycle or multiple periods instead of desired one cycle, a lock detector, having a wide operating range is employed. It works as a coarse phase detector by outputting \( V_{\text{under}} \) and \( V_{\text{over}} \) according to whether the delay is too small or too large. The operating principle of the lock detector is shown in Fig. 4. The delay line input is denoted \( P_0 \), and the output of the \( n \)-th delay stage \( P_{n+1} \). By looking at \( S_1 \) and \( S_2 \) at the rising edges of \( P_0 \), the total delay can be coarsely compared to the period of the reference. For correct locking, \( S_{2} = 0 \) and \( S_{2} = 1 \) (\( S_{1} \) and \( S_{2} \) are the values of \( S_{1} \) and \( S_{2} \) sampled at rising edges of \( P_0 \)), and neither of \( V_{\text{over}} \) and \( V_{\text{under}} \) is enabled (\( V_{\text{over}}=1 \), \( V_{\text{under}}=1 \)). If \( S_{2} = 1 \), the total delay is larger than \( N/(N+1) \) times of the period of the reference and the signal \( V_{\text{over}} \) is enabled (\( V_{\text{over}}=0 \)). If \( S_{2} = 0 \) and \( S_{2} = 1 \) are both zero, the total delay is smaller than \( N/(N+1) \) times of the period of the reference and the signal \( V_{\text{under}} \) is enabled (\( V_{\text{under}}=0 \)). The schematic of the lock detector is shown in Fig. 5. The operating range of this lock detector is from 0 to \( NT \) (total delay of the
delay line), which is larger than the tuning range of most delay lines. The signal $V_{\text{win}}$, a negative pulse around each rising edge of $P_0$ when neither of $V_{\text{under}}$ and $V_{\text{over}}$ is enabled, is used by the charge pump phase comparator described below.

**Charge Pump Phase Comparator**

The charge pump phase comparator is a critical component in this design, since it determines the in-lock error, which should be as near to zero as possible so that (3) might be satisfied. Most conventional phase detectors are designed for use in PLLs, where the in-lock error is not so important as long as it is constant. They are therefore not ideal for DLLs. In this design, a novel charge pump phase comparator is employed to obtain high resolution and reduce the in-lock error.

Fig. 6 shows the block diagram of the charge pump phase comparator. Signal $V_{\text{win}}$ coming from the window generator acts as a “clock” to synchronize operations of the charge pump phase comparator. With signal $V_{\text{win}}$ high, the circuit is in 'waiting state' thus pulse generator outputs $P_u$ and $P_d$ are both high, two integrators (IntU and IntD) are in a reset state thus $V_u$ and $V_d$ are both zero, and voltage comparator (VC) exhibits a high output impedance. Once signal $V_{\text{win}}$ goes low, the pulse generator (PG) produces two negative pulses $P_u$ and $P_d$ whose falling edges correspond to the rising edges of the $P_0$ and $P_N$ respectively and whose rising edges occur simultaneously when both of $P_0$ and $P_N$ go high. The phase difference between $P_0$ and $P_N$ is thus converted to the voltage difference between $V_u$ and $V_d$ obtained by integrating $P_u$ and $P_d$ separately. With $V_u$ and $V_d$, a voltage comparator produces an output current $I_{\text{out}}$ until $V_{\text{win}}$ goes high again. If either $V_{\text{under}}$ or $V_{\text{over}}$ is enabled, signal $V_{\text{win}}$ is kept high and the lock detector takes control to generate the output current $I_{\text{out}}$. To ensure that the voltage comparator works properly and that the high sensitivity of this charge pump phase comparator is maintained, the lower one of $V_u$ and $V_d$ must be slightly larger than the NMOS transistor threshold voltage.

This is achieved with the pulse control feedback circuit (PCFB).

The schematic of the pulse generator is shown in Fig. 7. This circuit acts on its inputs $P_0$ and $P_N$ if the signal $V_{\text{win}}$ is low. Its outputs $P_u$ and $P_d$ are two negative pulses whose widths and amplitudes correspond to the phases of its two inputs $P_0$ and $P_N$. Simultaneous amplitude adjustment of the two pulses is done using the signal $V_{\text{ctl}}$ which originates in the pulse control feedback circuit.

The schematic of remaining part of the charge pump phase comparator is shown in Fig. 8. The paths from $P_0$ to $I_{\text{out}}$ and from $P_N$ to $I_{\text{out}}$ are very short and symmetric except at the voltage comparator, so it can provide high resolution and low in-lock error even without carefully sizing the transistors.

**Simulation Results**

To verify the proposed architecture described above, an $N=9$ DLL-based frequency synthesizer was specified using 0.18μm CMOS technology. The post-layout simulation was done and the in-lock error and period jitter were measured as shown in Fig. 9. The RMS value of in-lock error is 0.1ps and that of the output jitter is 0.7ps.

![Fig. 6 Operating principle of charge pump phase comparator](image)

![Fig. 7 Schematic of pulse generator](image)

![Fig. 8 Schematic of charge pump](image)
Experimental Results
The 9-time frequency synthesizer described above has been fabricated and tested. It works with an input frequency range extending from 110MHz to 230MHz. The output phase noise produced when the device was operating with a 173.3MHz input (from RF signal generator) is shown Fig. 10. At 10kHz offset, the phase noise is -107.17dBc/Hz, which can be expected to be much lower if a better reference source such as a crystal oscillator was employed. The spur power was measured as shown in Fig. 11, which is -46.17dBc. Due to the DLL nature and the edge combining structure, the centre frequency is very stable comparing to that of PLL-based frequency synthesizers. Table I summarizes the specifications of the frequency synthesizer.

Conclusion
An integer-N DLL-based frequency synthesizer, employing a novel charge pump phase comparator and a novel delay line and edge combining circuit to greatly reduce the output spur and jitter has been presented. With a novel lock detection circuit, working with the charge pump phase comparator, this frequency synthesizer has a wide operating range and can always lock correctly without setting the initial control voltage. A 9-time frequency synthesizer has been designed and implemented. The measurement results show that its phase noise performance and spur performance are better than those of other reported DLL-based frequency synthesizers.

<table>
<thead>
<tr>
<th>CHARACTERISTICS OF MEASURED SYNTHESIZER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply voltage</td>
</tr>
<tr>
<td>Reference frequency</td>
</tr>
<tr>
<td>Output frequency</td>
</tr>
<tr>
<td>Current consumption</td>
</tr>
<tr>
<td>Phase noise @10kHz offset</td>
</tr>
<tr>
<td>Spur power</td>
</tr>
<tr>
<td>Input frequency range</td>
</tr>
<tr>
<td>Output frequency range</td>
</tr>
<tr>
<td>Process</td>
</tr>
<tr>
<td>Active area</td>
</tr>
</tbody>
</table>

Acknowledgment
A financial support from CITO and NSERC is gratefully acknowledged.

References