A New Approach to Fractional-N PLL Design Yields Performance Breakthrough

by Brendan Daly, Analog Devices

Fractional-N phase locked loops (PLLs) promise a theoretical performance edge over integer-N, but historically they have struggled to achieve this breakthrough. Now, a new design has resulted in a fractional-N PLL that lives up to its theoretical performance.

This article will discuss the limitations of existing architectures and their effect on the performance. It then discusses the motivation for developing fractional-N PLLs, their evolution and how the performance breakthrough was achieved. The ADF4252 fractional-N will be used to demonstrate the level of performance now attainable using fractional-N PLLs (**Figure 1**). A 1.8 GHz Local Oscillator (LO) application will be the basis for the practical examples.

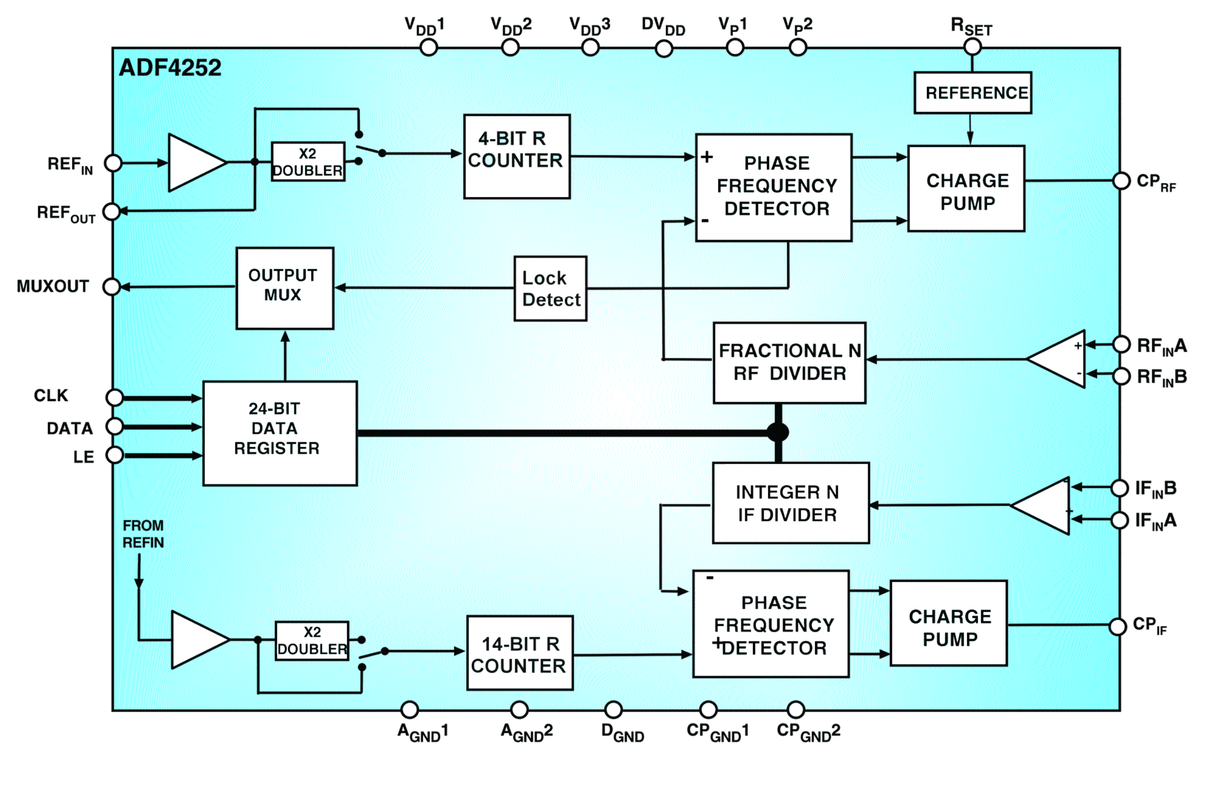

Figure 1: ADF4252 Functional Block Diagram

Integer N Versus Fractional N

Integer-N PLLs are governed by the following equation, which relates the RF frequency (RF_{OUT}) to the frequency at the phase-frequency detector (F_{PFD}):

$$N = RF_{OUT} / F_{PFD} \qquad (1)$$

To attain a channel step resolution on the RF output of 200 kHz, the PFD frequency (F_{PFD}) must be set to 200 kHz. This is because the N counter can only increment in integer values. Taking the LO example, N = 1800 MHz/200 kHz = 9000. To generate the next adjacent channel, 200 kHz away, N is incremented to 9001.

In a fractional-N PLL, the N divider is broken up into the integer (INT) divider and a modulus (MOD) divider, which acts as the fraction. The average division factor is now INT + FRAC/MOD, where 0 < FRAC < MOD.

$$(INT + FRAC/MOD) = RF_{OUT} / F_{PFD} \qquad (2)$$

This is the essence of fractional-N. Now the PFD frequency can be larger than the RF channel resolution. This reduces the phase noise, as will be explained below, but can cause degradation of the spurious performance. Use of a third-order sigma-delta based interpolator in the fractional engine–as is done in the ADF4252–provides improved phase noise performance without sacrificing spur performance.

Reference Spurs

One major stigma attached to fractional-N PLLs is their poor spurious performance. The ADF4252, however, uses the inherent dither ability of the sigma-delta to push the spurs down into the noise floor. When operating in *Lowest Spur* mode, the dithering circuit is enabled. This randomizes the discrete spurious energy, effectively turning it into white noise. Because the spurs are created digitally, the sigma-delta based fractional interpolator provides spurious performance that is extremely stable over temperature. Other available fractional-N solutions, which use analog fractional compensation, are subject to large variations in spurious performance. The ability of analog compensation to operate effectively is subject to temperature. The spurious performance variation of the ADF4252 over temperature will generally be less than 3 dB.

Programmable Modulus

Another significant step to improving performance is in the idea of the programmable modulus, which comes from a very simple characteristic of fractional-N PLLs. The spur nearest the carrier will appear at a frequency offset equal to the PFD frequency (F_{PFD}) divided by the modulus (MOD). Other PLLs offer very high resolution fixed modulus values, e.g. 2^{18} or 2^{22}. Put that into a real application and the limitations become apparent very quickly. Revisiting the example, the PFD is run at 13 MHz. If the fractional-N PLL has a fixed 22-bit modulus, it follows that the first spur will appear at 13 MHz/2^{22} = 3 Hz offset from the carrier. There will also be harmonics of this spur, so spurs will appear at 6 Hz, 9 Hz, 12 Hz etc. In a GSM1800 application, the bandwidth of the loop filter is typically chosen to be roughly 20 kHz. This means that the spurs see no attenuation when passing through the low pass loop filter, and appear on the RF output of the system. In fact, because the spurs are so close together, they actually look like broadband noise, degrading the overall performance of the system.

The user might think that there is a noise contribution from somewhere, but it is in fact the forest of spurs. **Figure 2** shows this. The noise floor is -90 dBc/Hz, but the spurs turn this into an effective noise floor of -70 dBc/Hz. This is the inherent liability to a fixed modulus approach. A high level of resolution is required to achieve frequency accuracy. Compare the fixed modulus approach with a solution using a programmable modulus. With the ADF4252 in the example, the modulus can be programmed to 65, to generate the necessary 200 kHz channel steps from a 13-MHz PFD frequency. From above, it follows that the spur will appear at 13 MHz/65 = 200 kHz offset from the carrier. A 20-kHz loop bandwidth will provide one decade of attenuation on this 200-kHz spur. Therefore, the spur using a programmable modulus will be much further from the carrier, and will be attenuated by the loop filter. Its harmonics are much further apart, and will also be attenuated by the filter. This means there is no broadband noise associated with the fixed modulus approach. As can be seen in **Figure 3**, the in-band noise performance is -90 dBc/Hz.

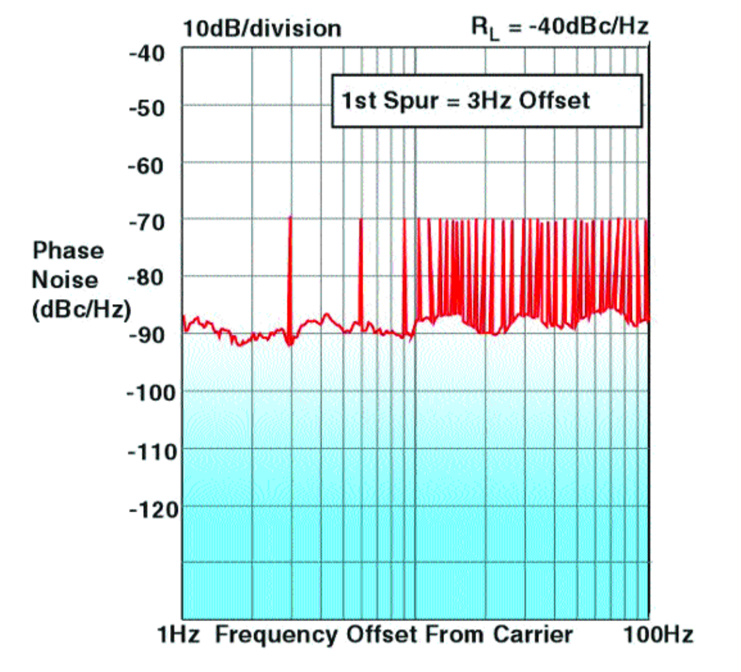

Figure 2: Spurious Contribution of a High Resolution Fixed Modulus

Fixed vs. Programmable Modulus

It is clear that a programmable modulus is a more effective way of implementing a fractional-N solution for spurious performance. The next question to ask is why the programmable modulus hasn't been available before? The logic that applied to designing with a fixed modulus was this: To achieve the RF output frequency as accurately as possible, the modulus must be made as large as possible for increased resolution. Referring back to the example, to synthesize 1.8 GHz RF_{OUT}:

$$RF_{OUT} = R \_ [INT + (FRAC/2^N)]$$

(N is the resolution of the modulus, say 18 bits)

$$RF_{OUT} = 13\ MHz \_ [138 + (60495/2^{18})$$

$$RF_{OUT} = 1.800000023\ GHz.$$

An 18-bit modulus can achieve accuracy to within

Analog Devices, Con't on pg 28

Analog Devices, Con't from pg 18

23 Hz. Using a fixed 22-bit modulus allows the PLL to synthesize frequencies to within 3 Hz of the desired frequency. Therefore the resolution of the modulus is made as large as possible to make the RF frequency more accurate. But, incorporating a programmable modulus does away with the need for very high resolution. This is a key aspect that had been previously overlooked. The ADF4252 has programmable 12-bit resolution. This means that any modulus from 1 to 4095 can be used. Referring to the example again, to realize 1.8 GHz output, with 200 kHz resolution and a 13 MHz reference source, the modulus is programmed to 65. The output frequency is again governed by the equation:

RF = Fpfd _ [INT + (FRAC/MOD)]

By programming the modulus to 65,
RF$_{OUT}$ = R _ [INT + (FRAC/MOD)]
RF$_{OUT}$ = 13 MHz . [138 + (30/65)]
RF$_{OUT}$ = 1.8 GHz

The programmable modulus ensures the correct frequency will be realized. Therefore a very high order fixed modulus is unnecessary.

Fixed vs. Programmable Modulus

The idea of having a programmable modulus adds huge flexibility of operation for the user. The programmable modulus is very useful for multi-standard applications. The ADF4252 has 12 bits of resolution, but the fact that it is completely programmable makes it flexible. There are yet more advantages to having a programmable modulus. It can ensure loop stability is maintained for multi-standard applications. If a dual-mode phone requires PDC and GSM1800 standards, the programmable modulus is beneficial. PDC requires 25 kHz channel step resolution, whereas GSM1800 requires 200 kHz channel step resolution. A 13 MHz reference signal could be fed directly to the PFD. The modulus would be programmed to 520 when in PDC mode (13 MHz / 520 = 25 kHz). The modulus would be reprogrammed to 65 for GSM1800 operation (13 MHz / 65 = 200 kHz). The important factor is that the PFD frequency remains constant (13 MHz). This allows the user to design one loop filter, which can be used in both setups without running into stability issues. It is N, the ratio of the RF frequency to the PFD frequency, that affects the transfer function, and hence stability of the system.

$$Closed\_Loop\_Gain = \frac{K_d.K_v.Z(s)}{N.s}$$

Keeping this N relationship constant and instead changing the modulus factor results in a stable filter.

Phase Noise

The ADF4252 offers substantial improvements in phase noise performance over existing integer-N PLLs. In a GSM1800 application, the in-band phase noise improvement can be as much as 15 dBc/Hz. The reason is as follows: The in-band phase noise using an integer-N PLL is the noise floor of the PFD, degraded by 20 log N (where N is the ratio of RF frequency to the PFD frequency). In our example, N = 1.8 GHz/200 kHz = 9000. This means that the noise at the VCO output is the PFD noise, degraded by 20 log 9000 = 78 dB. In Fractional-N, a much higher PFD frequency is used. So the 20 log N degradation is not so severe. Returning to the example, N = 138, therefore, the PFD noise is only degraded by 20 log 138 = 43 dB. There is a penalty for operating the PFD at a higher frequency. This penalty is the 10 log (PFD Frequency increase) = 10 log (13 MHz/200 kHz) = 18 dB. So the net phase noise improvement is 78 - 43 - 18 = 17 dB.

Figure 4 shows a measured comparison of the phase noise performance between the ADF4252 fractional-N and an integer-N PLL for a GSM1800 setup. Until now, fractional-N solutions have promised this level of performance in theory, but failed to transfer this improvement onto silicon. The reason was that the fractional engine being used was contributing noise to the system–so much so that it was negating the theoretical improvement expected. Some of this noise contribution was coming from the broadband noise, as described above, so the programmable modulus was just part of the plan to improve the noise performance. The realization that broadband noise was one of the key issues that needed to be resolved, plus extensive knowledge in sigma-delta technology, led to the breakthrough of using dithering and noise shaping in the fractional engine. A third-order sigma-delta modulator was determined to provide the optimum performance in the ADF4252. To improve phase noise even further, particular focus was given to the charge pump linearity because nonlinearity of the charge pump translates directly into spurious energy. Minimizing the spurs in the first place meant that the sigma-delta had less dithering to perform, and hence, less white noise resulted. The ADF4252 is a general-purpose part, so it was decided to offer software programmability so that the user can make the noise vs. spurious performance tradeoff, allowing the user to optimize the ADF4252 for the application and specifications.

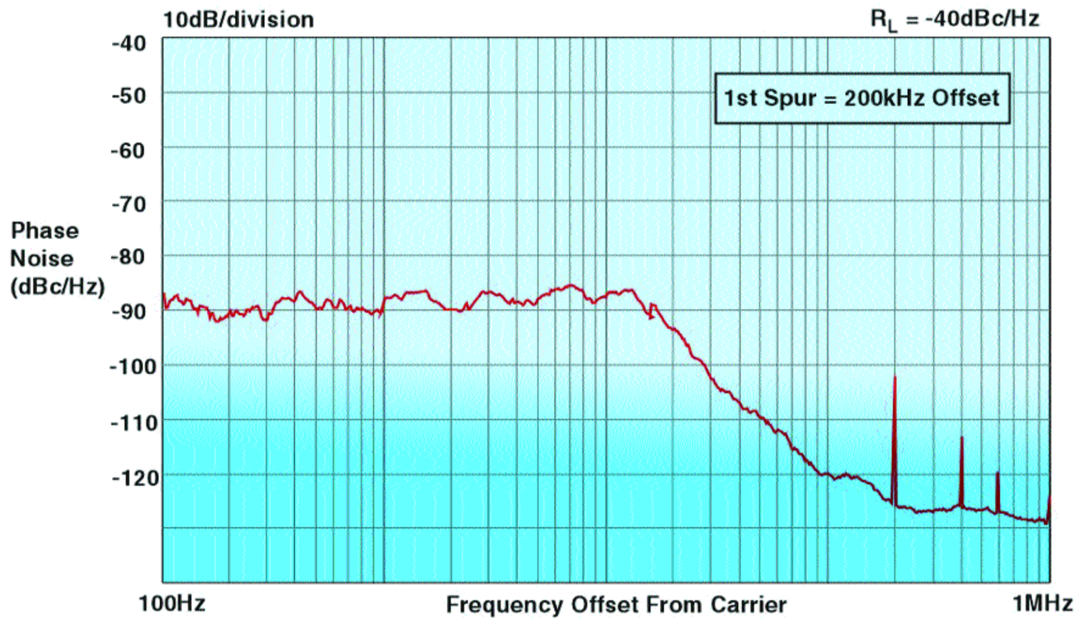

Figure 3: Spurious Contribution of a Programmable Modulus

Phase Re-sync

Phase re-sync is another useful feature in the ADF4252. Next generation wireless infrastructure is looking at the issue of frequency reuse in adjacent cells. This would be necessary to increase capacity. In GSM at present, adjacent cells do not operate at the same frequencies, due to interference that can occur between adjacent antennas transmitting on the same frequency. This bandwidth reduction inherently limits the capacity of the system. Fractional-N PLLs can be used to phase the antenna array, so that adjacent cells can reuse the same frequency. The idea is that the phase of one antenna would interfere with the phase of another antenna within its own cell. It would provide constructive interference (in-phase) in certain directions, and destructive interference in other directions. This would ensure that frequency use in adjacent cells is possible. The issue is that if a fractional-N PLL jumps from frequency A (and phase A) to frequency B (and phase B), then back again to frequency A, it will not necessarily return to phase A. It can return to any of M phases, where M is the value of the modulus (MOD). So the ADF4252 in the worked example could have 65 possible phase states. The phase Re-sync feature solves this problem. The re-sync counter sets the number of PFD cycles before the sigma-delta is reset, after a new channel has been programmed. Resetting the sigma-delta integrators to their seed values every time a new channel is programmed ensures that the synthesizer always returns to the same phase.

Conclusion

In conclusion, the ADF4252 takes a fresh approach to a fractional-N PLL solution, with the focus firmly on performance and adaptability. By paying particular attention to the way in which the sigma-delta based fractional engine and charge pump affect the PLLperformance, this flexibility can be offered as a programmable optimization solution, rather than a fixed design.

Bibliography

1. *ADF4252 Fractional-N PLL Data Sheet*, Analog Devices Inc.
2. *ADF4110/11/12/13 RF PLL Frequency Synthesizers'Data Sheet*, Analog Devices Inc.
3. *Phase Locked Loops 3rd Edition,* Roland E. Best.
4. *PLL Performance, Simulation and Design,* Dean Banerjee.
5. "Phase-locked loops for high-frequency receivers and transmitters", *Analog Dialogue,* (March 99), Mike Curtin & Paul O'Brien.

ANALOG DEVICES

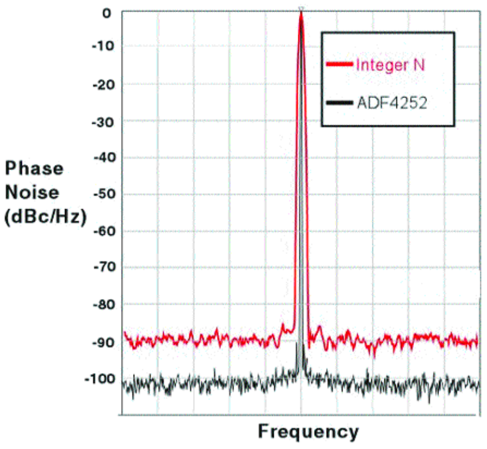

Figure 4: In-Band Phase Noise Comparison between Integer-N and Fractional-N