Chapter 8

POLYPHASE MULTIPATH CIRCUITS FOR COGNITIVE RADIO AND FLEXIBLE MULTI-PHASE CLOCK GENERATION

Eric A. M. Klumperink, Xiang Gao and Bram Nauta
IC-Design Group, CTIT, University of Twente, Enschede, The Netherlands

Abstract: In this chapter we discuss flexible cognitive radio circuits for dynamic access of unused spectrum. Ideally, such circuits can work at an arbitrary radio frequency (RF). We review techniques to realize radios without resorting to frequency selective dedicated filters [24], in particular a recently proposed polyphase multipath technique canceling harmonics and sidebands [11,12]. Using this technique, a wideband and flexible power upconverter with a clean output spectrum can be realized on a CMOS chip, aiming at flexible radio transmitter applications. Prototype chips can transmit at an arbitrary frequency between DC and 2.4GHz. Unwanted harmonics and sidebands are more than 40dB lower then the desired signal up to the 17th harmonic of the transmit frequency. Such polyphase multipath circuits need flexible multi-phase clocking with a large frequency range and low phase errors. We will compare a Shift Register (SR) to a Delay Locked Loop (DLL) for multi-phase clock generation, and motivate why a SR is not only more flexible but often also better [16]. For a given power budget, we show that a SR almost always generates less jitter than a DLL, assuming both are realized with current mode logic. This is due to differences in jitter accumulation and the possibility to choose latch delays in a SR much smaller than the delays of DLL elements. For *N*-phase clock generation, a SR also functions as a divide-by-*N* and requires a VCO with *N* times higher frequency. However, this does not necessarily lead to more power consumption and can even have advantages like higher Q and less area for the inductors.

Key words: Cognitive radio, Software defined radio, Radio transceivers\, Harmonic Rejection Mixer, Jitter, Nonlinearity, CMOS, Clock Generation, Multi-Phase Clocks, Current Mode Logic, Delay Locked Loop, DLL, Divider, Jitter, Timing Jitter, Phase Noise, Shift Register

1. INTRODUCTION

Cognitive radios aim at exploiting the scarcely available radio spectrum in a smart flexible way. Traditional TV bands between 50MHz and 900MHz are currently being freed for new applications. New licensed users are planned (e.g. DVB-H), but in addition new ideas for more flexible use of the spectrum are explored [1]. For higher frequencies similar ideas are developed. In general, regulatory organizations seem to move in the direction of providing more freedom to new standards, where only a minimum set of requirements are enforced. E.g. regulations might allow to exploit white spectrum, where "Detect And Avoid" rules are defined (e.g. response times, maximum interference levels to incumbent services). This will lead to new radio systems with different requirements on the radio software and hardware. In this chapter we will mainly focus on the impact of cognitive radio system requirements on the physical layer (PHY), and especially the radio frequency hardware. Flexible multi-phase clocking will turn out to play a crucial role, and will be discussed in detail.

To allow for flexible spectrum access, a flexible radio hardware platform is desired, allowing for flexible choice of the radio frequency depending on free available spectrum. Traditional radio hardware is primarily optimized for cost and low power, but not for flexibility. Low power is often achieved using inductors and capacitors in resonating circuits with a high quality factor, dissipating only a fraction of the maximum energy stored in the reactive components. However, such circuits only work effectively in a narrow band around their resonance frequency, and are hence application specific for a certain band. Micro-Electrical-Mechanical system (MEMs) technology may help to relax this problem; however for reasons of cost and form factor fully integrated solutions in mainstream CMOS technology are preferred if feasible. Thus we focus in this chapter on CMOS circuits and IC architectures. We will analyze the desired functionality of the radio interface for dynamic spectrum access, and look at some feasibility bottlenecks induced by CMOS circuit properties, like timing jitter, nonlinearity and time-variance. Some possible solution directions are reviewed, especially a recently proposed polyphase multipath technique. This technique allows for realizing a highly flexible radio transmitter for the DC-2.4GHz range on a CMOS chip without dedicated filters. It requires multi-phase clocks for which the phase-accuracy is critical. Two competing techniques to realize such clocks, one based on a Shift Register (SR) and the other on a Delay Locked Loop (DLL), are discussed in the second half of this chapter, to show that SR-based clocking has fundamental advantages.

2. FLEXIBLE RX/RFS: NOT JUST AN ADC

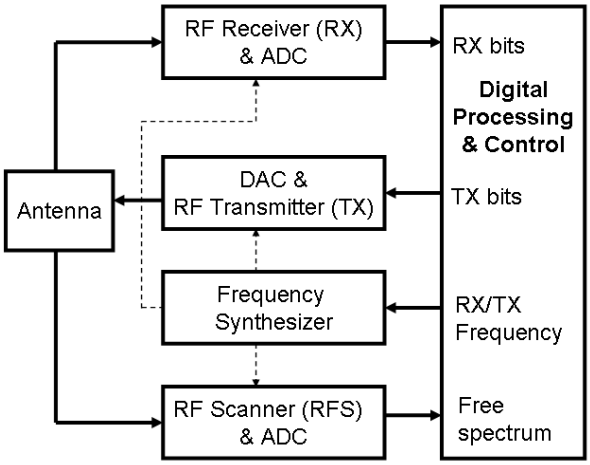

Figure 8-1. Block diagram of a cognitive radio system for dynamic spectrum access.

Fig. 8-1 shows a high level functional block schematic of a cognitive radio. It consists of an antenna connected to a radio receiver (RX), a radio transmitter (TX) and a Radio Frequency Scanner (RFS). A Baseband Processing and Control unit processes the spectral information, and decides which frequency is free for use. It controls the frequency synthesizer to generate the desired radio frequency carrier, sends bits to the TX and receives bits from the RX.

Ideally, a cognitive radio should be able to communicate wherever free spectrum is available, i.e. be very flexible in terms of the transmit frequency. This suggests a wideband radio receiver should be used for detecting free spectrum and receiving data, in contrast to traditional narrowband radio systems. For maximum flexibility, radio signal processing should be done in the digital domain. On a high abstraction level, a cognitive radio can then be considered as an A/D Converter (ADC) for the RX and RF Scanner blocks, and a D/A Converter (DAC) for the TX block.

To judge the feasibility of a wideband ADC based receiver, data from Walden's overview paper on ADCs is useful [2]. Consider a mobile radio communication receiver operating at popular radio frequencies between 0.05-6GHz. Typical transmit power levels for mobile radio standards are in the range of 10mW up to more than 1W. The radio path-loss strongly varies

from case to case, but it is quite common to receive radio antenna voltages in the range from 1μV up to 100mV. To detect a weak 1μV signal, in the presence of a 100mV interferer, we need an ADC with more than 100mV/1μV=100.000 detection levels, i.e. roughly 2^{16} levels (16 bits). To observe 5GHz signals, the ADC should at least take 10 Giga samples every second. Assuming for a moment this is technically feasible, at a (rather optimistic) energy of 1pJ per conversion [2], this leads to a power consumption of 10^{10} samples/second × 2^{16} levels × 10^{-12} J ≈ 1kW! The energy per conversion decreases only slowly over time because analog accuracy requirements are involved, which do not benefit much from Moore's law. Note also that the actual radio bandwidth of interest is typically orders of magnitude lower then the radio-carrier frequency. This makes "full-Nyquist" A/D conversion really overkill, and a waste of power, even if it would become technically feasible. Thus we feel there is a need for architectural innovations to make highly flexible cognitive radio systems feasible.

A more realistic and still reasonably flexible approach is to down-convert an RF signal of interest to DC ("zero-IF architecture"), reduce its bandwidth and dynamic range by low-pass filtering and then do the A/D conversion at a rate and a resolution which are feasible at 10-100mW A/D converter power. Recently a software defined front-end using this approach for the 500MHz-5GHz band has been proposed [3]. It uses a wideband low noise amplifier exploiting thermal noise cancellation [4], followed by a highly linear passive down-conversion mixer. However, as there is hardly any RF pre-filtering, the linearity requirements on the RF front-end are very high. Moreover, wideband down-converters using hard-switched mixers are plagued by spurious responses, i.e. they do not only down-convert the wanted RF-band, but also its harmonics. Thus harmonic rejection mixers are needed, e.g. as proposed in [3,5]. We will address this harmonic rejection mixing later in this chapter when dealing with upconversion mixers.

3. SAMPLING CLOCK JITTER REQUIREMENTS

Instead of a mixer, a sampler can also be used for frequency down-conversion. Whereas full Nyquist rate A/D conversion of GHz signals is currently far from feasible, sampling at GHz rates *without* high resolution *quantization is* practical, as demonstrated for a Bluetooth and GSM receiver [6]. These receivers sample the antenna signal at RF and then process it in the charge domain via passive switched capacitor circuits. Via decimation with internal anti-alias filtering, the sample rate is reduced to a sufficiently low rate to do A/D conversion at acceptable power consumption [6].

The sampling at RF might surprise people who work on low jitter sampling clocks for high-speed ADCs, where clock jitter requirements are increasingly becoming a feasibility bottleneck. This is because timing uncertainty shifts the sampling moments, introducing significant amplitude errors especially for high-amplitude high-frequency signals. To keep these errors from degrading the resolution of the ADC, an extremely low RMS-jitter of less than 11 fsec would be needed for an 11 bit ADC sampling a 6GHz full swing sine wave signal [7].

Fortunately, for radio receiver applications, sampling jitter turns out to be much less harmful. This is because radio signals are narrowband in nature, so *only the noise level in the wanted channel band* is relevant. Jitter in a sampling clock introduces noise at the output of the sampler which strongly varies with frequency and is mainly concentrated around strong high-frequency interferers [7]. The roll-off with frequency distance from the interferer depends on the shape of the phase noise spectrum of the sampling clock. Overall, the requirement on the sampling clock jitter is close to what is needed for traditional mixer based receiver systems limited by reciprocal mixing [7]. Calculation for a Bluetooth receiver shows that 1.3psec RMS-jitter can be accepted, which is more than *two orders of magnitude easier than corresponding ADC clock jitter specs* [7]. Thus jitter is not as big a problem as often thought, opening the door for radio architectures exploiting high-speed sampling like in [6]. Still, if no or not enough RF-filtering is used, RF signals at harmonics of the sampling clock will again be downconverted and will interfere with the desired signal. Thus harmonic rejection techniques are needed, e.g. as proposed in [8].

4. FLEXIBLE TX: NOT JUST A DAC

Realizing a flexible transmitter using a DAC seems possible in principle, as the dynamic range of a transmitted signal is typically significantly lower than the dynamic range of a received signal. However, apart from the useful TX-signal, many other spurious components may be produced. As an integrated radio transmitter should produce significant output power, typically in the range of milli-Watts up to a few Watt, power drivers and power amplifier circuits with transistors working at large signal swings are used. Thus non-linearity of the transistors plays an important role, resulting in harmonics (see Fig. 8-2) and intermodulation distortion products at many unwanted frequencies [9]. As the power efficiency of most amplifiers increases for higher signal swings, it is desirable to drive the amplifiers to a level close to their compression point. However, in practice significant "back

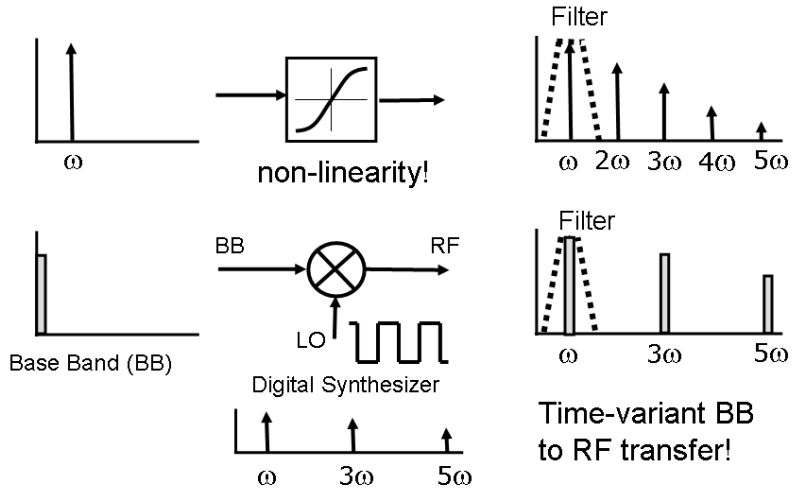

Figure 8-2. Nonlinearity and time-variance due to switched mixers generate unwanted spectral components, which are traditionally removed by dedicated band-pass filters.

off" is needed [10] to suppress distortion products sufficiently at the cost of efficiency.

Apart from nonlinearity, a time-variant transfer function can also introduce many unwanted frequency components. Ideal DACs and hard-switched mixers can be modeled as linear time-variant circuits, with a linear transfer from input to output, which changes instantaneously with the state of the clock signal. For simplicity, we only discuss the case of an upconversion TX-mixer here, but similar conclusions hold for a DAC. The mixer is shown in Fig. 8-2, where an ideal 50% square wave switching between +1 and -1 models the hard-switching mixer operation. This square waveform has odd harmonics with a relative strength of 1/3, 1/5, 1/7, etc. compared to the fundamental. Thus the 9^{th} harmonic is still stronger than -20dB compared to the fundamental.

In order to avoid harmonic mixing, the input signal could be multiplied by a sine wave signal using a highly linear multiplier. However, realizing a linear multiplier is much more difficult then a hard-switched mixer, and the generation of a clean sine wave is problematic, especially when a large frequency range is involved. Typical sine-wave oscillators, e.g. LC oscillators have only a limited tuning range in the order of 5-50%. If a larger tuning range is needed, digital dividers are commonly used to divide the VCO frequency to an appropriate value. As digital circuits benefit from Moore's law, we strongly prefer flexible digital synthesizer techniques over

analog sine wave generation. However, this means we have to find a solution to suppress unwanted harmonics.

In traditional radio transmitters, these unwanted products are rejected using dedicated band-pass filters typically implemented using inductors and capacitors (LC filters). We like to avoid such filters on CMOS chips, as they require high quality inductors which are difficult to implement and/or take large chip area. For dynamic spectrum access, such filters are even more problematic as LC band-pass filters work at a fixed frequency related to the LC-resonance frequency, which limits the flexibility in choosing a TX-frequency. The next section discusses a recently proposed polyphase multipath technique to eliminate these filters or relax their requirements significantly.

5. POLYPHASE MULTIPATH CIRCUITS FOR SPECTRAL PURITY ENHANCEMENT

Fig. 8-2 shows a nonlinear circuit excited by a single sine wave at ω, producing a wanted output signal at ω but also unwanted harmonic distortion at 2ω, 3ω, 4ω, etc.. Fig. 8-3 shows a polyphase 3-path circuit, cancelling many harmonics of ω [11]. The basic idea is to divide a nonlinear circuit of Fig. 8-2 into 'n' equal smaller pieces, and apply an equal but opposite phase shift before and after each nonlinear circuit. If the phase shift in path 'i' is (i-1)×φ, where φ is a phase shift constant satisfying n×φ=360°, the circuit will produce the same wanted harmonic as Fig. 8-2, but cancel many higher harmonics. Mathematically this can easily be shown using a power series expansion, assuming a memory-less weakly nonlinear system. If the signal x(t) = Acos(ωt) is applied to the input, the output of the nonlinear circuit of the i^{th} path can be written as:

$$\begin{aligned} p_i(t) = a_0 &+ a_1\cos(\omega t + (i-1)\varphi) \\ &+ a_2\cos(2\omega t + 2(i-1)\varphi) \\ &+ a_3\cos(3\omega t + 3(i-1)\varphi) + ... \end{aligned} \quad (1)$$

Where a_0, a_1, a_2, a_3…. are Taylor series constants characterizing the nonlinearity [9]. From Eq. (1), it can be seen that the phase of the 'k^{th}' harmonic at the output of the nonlinear circuit rotates by 'k' times the input phase (i-1)φ. The phase shifters, -(i-1)φ, after the nonlinear blocks are required to align the fundamental components at ω in phase again.

The signals at the output of these phase shifters can be written as:

$$y_i(t) = a_0 + a_1\cos(\omega t) + a_2\cos(2\omega t + (i-1)\varphi) + a_3\cos(3\omega t + 2(i-1)\varphi) + \ldots\ldots \quad (2)$$

In Eq. (2), the phase of the fundamental component is identical for all the paths, but the phases of the harmonics are different for each path. If the phase φ is chosen such that $\varphi=360^o/n$, then all the higher harmonics are cancelled [11], except for the k^{th} harmonics for which k equals j×n+1 (j=0, 1, 2, 3, ..).

The simplest example of a polyphase multipath circuit is a well-known differential circuit driven with balanced (anti-phase) input signals. It cancels all even harmonics (no cancellation of k=j×2+1, i.e. odd harmonics).

A system with three paths is shown in Fig. 8-3. In this case, phase shifts of 0°, 120° and 240° are added before the nonlinear block to path 1, 2 and 3 respectively, and equal but opposite phases -0°, -120° and -240° behind the block. Due to the nonlinearity, the phase rotation for the k^{th} harmonic is k times the input phase. Thus the respective phases at the output of the nonlinear block for path [1,2,3] are [0°, 120°, 240°] for ω, [0°, 240°, 120°] for 2ω and [0°, 0°, 0°] for 3ω products. Fig. 8-3 also shows how the phases of the harmonics at the output of each path combine. Only the fundamental components add up in phase (red arrows), while the black and blue vectors for the second and third harmonics create a "balanced structure" at the output, resulting in a zero sum (cancellation). However, the fourth harmonic components will align in phase again, and will add up like the fundamental. The output spectrum in the lower part of Fig. 8-3 shows that the 2nd, 3rd, 5th, 6th etc harmonics are cancelled and the first non-cancelled is the fourth for a 3-path system. Similarly for a 4-path system the first non-cancelled harmonic will be the fifth harmonic and in general for an n-path system the $(n+1)^{th}$ harmonic is the first non-cancelled harmonic. Theoretically, an infinite number of paths is needed to cancel all the harmonics. However, in practice higher order harmonics are weaker than low order harmonics and need not all be cancelled. Also, some filtering will in practice always be present, e.g. due to the limited bandwidth of an antenna or the speed limitations in a circuit. Moreover mismatches will put a practical limit on what is feasible [11].

If the non-linear system is excited by a two-tone input signal $x(t) = A_1\cos\omega_1 t + A_2\cos\omega_2 t$, besides harmonics the output will also contain intermodulation products at new frequencies $p\omega_1+q\omega_2$, where p and q identify harmonics of ω_1 and ω_2 respectively, and can be positive or negative integer numbers. It can be shown easily that many intermodulation products are cancelled, except if p+q equals j×n+1 (where j=0, 1, 2, 3, …).

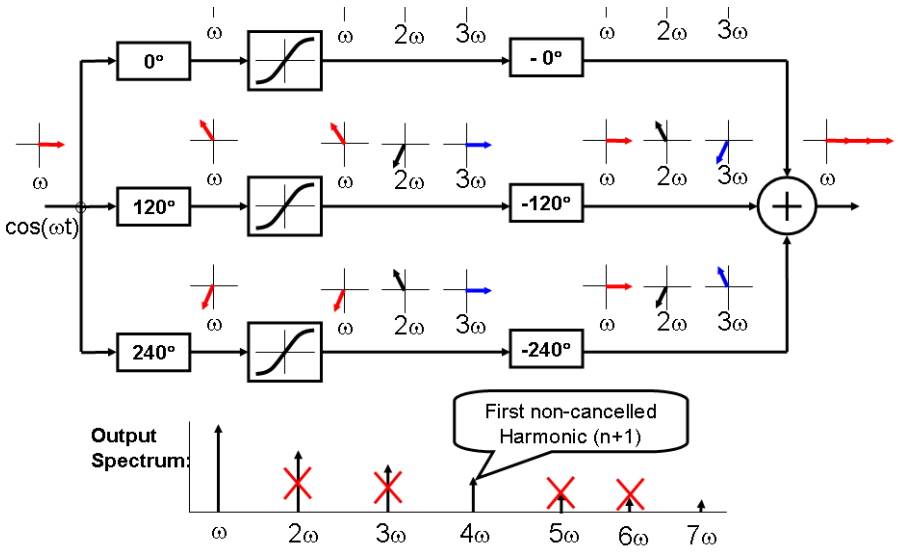

Figure 8-3. Polyphase 3-path circuit with harmonic cancellation except for harmonics j×n+1 (in this case n=3, so harmonics 1, 4, 7,.. are not cancelled)

6. MIXER: PHASE AND FREQUENCY SHIFTER

To realize wideband harmonic rejection using a polyphase multipath system, we need very wideband phase shifters before and after the nonlinearity. This is because all phase shifters need to have a constant phase shift over all relevant frequencies involved in the cancellation process. In a DSP intensive radio transmitter, digital signal processing techniques can be exploited to realize phase shifters before D/A conversion and nonlinear power amplification. Therefore, a good solution can be to shift this polyphase generation problem to the digital domain, and use a DSP followed by multiple DACs to generate multi-phase baseband signals. However, behind the nonlinear element we are in the analog domain, and there can be many harmonics. In that case cancellation of a multitude of harmonics requires a constant phase shift over many octaves of frequency.

A very wideband phase shifter can be implemented with a mixer, since a mixer as shown in Fig. 8-2 transfers phase information of both the "baseband" (BB) and "Local Oscillator" (LO) port to the output. Whatever phase is added to the LO signal will appear at the output of the mixer. So by replacing the second set of phase shifters in Fig. 8-3 with mixers, as shown in Figure 8-4, we can achieve a wideband phase shift but simultaneously we

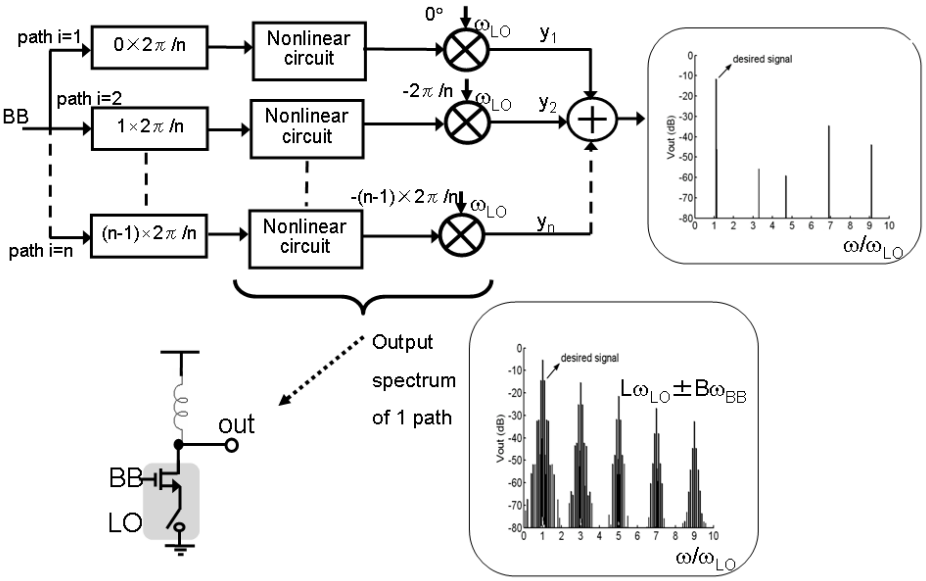

Figure 8-4. Polyphase n-path transmitter with mixers as 2nd phase shifters. Each path can be as simple as a switch and transistor, but produces many harmonics and sidebands due to time-variance and nonlinearity. The polyphase n-path system cancels most of these terms.

will get frequency conversion. As upconversion is desired in a transmitter circuit anyway, this fits nicely to our goal. However, a mixer produces not only a sum frequency but also a difference frequency. Usually only one of these is the wanted signal, while the other ("the image") needs to be suppressed. Moreover, the LO-signal usually is a square wave containing many harmonics, because flexible frequency synthesizers rely on digital dividers, as discussed in the previous section. For power efficiency reasons it is also highly desired to use a switching mixer and a large BB-signal swing, e.g. a single transistor with switch as shown in Fig. 8-4. Thus, the output spectrum for one path will now contain a forest of harmonics and sidebands as shown in the lower part of Fig. 8-4 for the case with a single-tone BB-signal. Spectral components occur at frequencies $L\omega_{LO} \pm B\omega_{BB}$, where L and B are integers, due to the multiplication of the square wave LO with the baseband input signal BB, and also the nonlinearity of the circuit. In the next section we will see how we can exploit the polyphase multipath technique to cancel almost all the unwanted components.

7. FILTER-LESS POWER UP-CONVERTER

A power upconverter combines the functionality of a power amplifier and upconversion mixer. The PA and mixer can be as simple as shown in Fig. 8-4, which is equivalent to first amplification and then mixing. Here the PA is a single transistor operating as transconductor (V-I converter), which is switched on and off by the LO signal via a switch (NMOS transistor driven by a digital inverter). Thus the V-I conversion and upconversion is done in the same circuit, via a switched transconductor mixer [13]. With respect to efficiency this circuit resembles a single transistor (class A) power amplifier. However, due to the polyphase multipath technique distortion products are cancelled and larger signal swings can be tolerated, improving efficiency.

Unfortunately, a few problematic products still remain present at the output. Since we have two input ports now (BB and LO), and mixing produces several sum and difference frequencies, a slightly different condition for non-cancelled products is found [11,12] ($L=j\times n+B$ where $j=\ldots.-2, -1, 0, 1, 2\ldots$, and B is a positive or negative integer number).

Especially the $3\omega_{LO}+3\omega_{BB}$ is troublesome because the 3rd order distortion term is usually much stronger than higher order distortion components [9] and is also close to the desired signal. It cannot be cancelled with any number of paths as all products for which L=B are not cancelled (j=0 case, so independent of n). To eliminate the strong $3\omega_{LO}+3\omega_{BB}$ terms, the duty cycle of the LO was chosen to be 1/3 [12]. By doing so, the 3rd, 6th, 9th, etc harmonic terms disappear from the Fourier series expansion, however some even order terms appear. Fortunately, it is quite easy to cancel even order products by using a differential baseband input (balancing).

To demonstrate the feasibility of a highly flexible multipath transmitter, we designed a power upconverter in a 0.13μm CMOS process, covering all frequencies up to 2.4GHz [12]. To show wideband spectral cleaning we designed an 18-path system, which can clean-up the spectrum up to the 17th harmonic. Fig. 8-5 shows the 18-path power upconverter. Each path consists of a switched transconductor mixer [13] with a baseband signal applied to a differential pair, acting as a differential transconductor (V-I converter), and an LO signal driving a grounded switch. The output currents of the V-I converters are easily added by connecting them together, and the wanted output signals from all paths add up in phase. Thus the total area and power of the power upconverter core is not increased by splitting it into 18 paths.

The V-I converter transistors are biased at the supply voltage via two large inductors (see Fig. 8-5) to increase the output swing and efficiency, as commonly done in power amplifier design. The inductance and the load

resistance constitutes a high-pass AC-coupling, which puts a lower limit to the RF frequency, but the chip itself can work at arbitrarily low frequency.

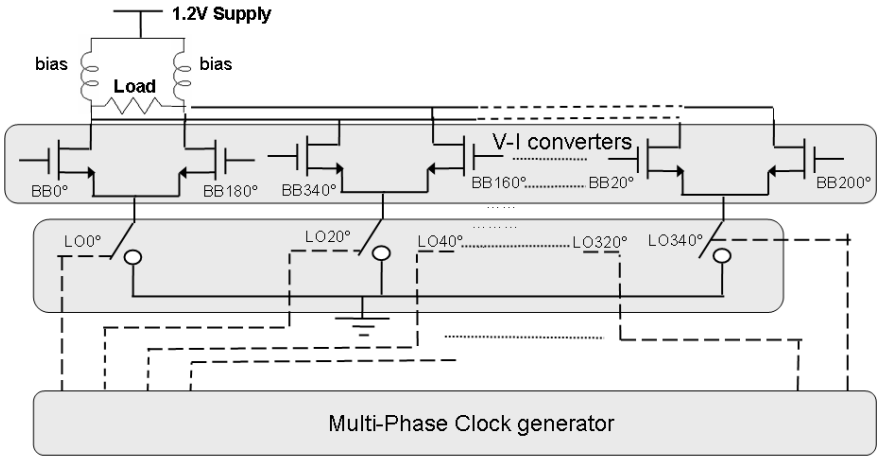

Figure 8-5. Circuit concept of an 18-path power upconverter [12]

Operating each individual switched transconductor mixer at the 1dB compression point, the upconverter is designed for a large output swing of about 2.5V differential peak-to-peak voltage, to maximize efficiency. This is close to the maximum swing that can be achieved from a 1.2V supply while keeping the output transistors in strong inversion and saturation, to maintain V-I converter functionality. For a 100ohm load, the 2.5V swing corresponds to roughly 8mW output power. To further increase the output power without adding an external power amplifier, a transformer could be added for broadband impedance transformation while scaling up the output current via wider transistors. To maximize the flexibility and frequency range, we implemented the LO phase generation via a current mode logic shift register running at 9 times the LO frequency. This enabled us to evaluate the circuit for an arbitrary LO-frequency between DC and a maximum given by the speed limitation of the logic used to realize the shift register. For 18 paths we need LO signals of 18 different phases (0°, 20°, 40° …340°) with 1/3 duty cycle. Applying a positive and a negative clock edge alternately to successive latches in a chain of 18 D latches (see Fig. 8-5), 18 different phases are produced. The feedback through the NOR gate is used to make the duty cycle 1/3.

In our experimental setup, the 9 differential baseband voltages with different phases are generated off-chip. More work has to be done to explore

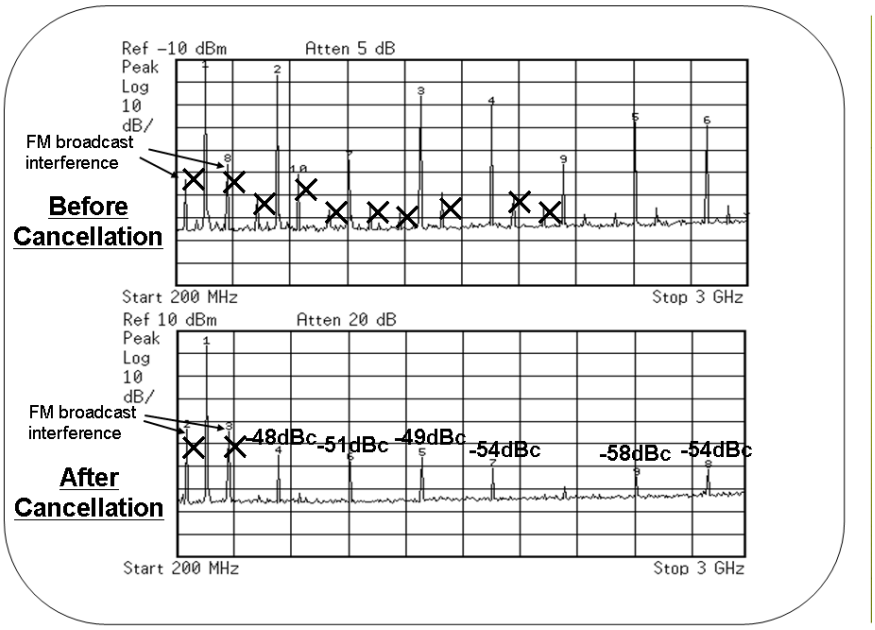

Figure 8-6. Output spectra of the 18-path Power Upconverter (PU) chip [12], with out-of-band power <-40dBc up to the 17th harmonic (LO=350MHz).

the most effective way to generate multi phase baseband signals on-chip via DSP techniques and multiple DACs.

The multipath technique cleans the output spectrum from unwanted harmonics, which result from the hard-switching mixer, but also from non-linearity in the switched transconductor. Simulations and measurements show that we can drive the power upconverter close to its 1dB compression point with harmonics well below <-40dBc and realize the high 2.5V output voltage swing directly over the load (e.g. antenna). Note that the two inductors are only used for biasing, and not for (dedicated) band-pass filtering.

The proposed upconverter has been fabricated in a 0.13µm CMOS process and takes an active area of only 0.14 mm^2. It delivers 8mW output power to a 100Ω off-chip load [12]. Fig. 8-6 shows the output frequency spectrum for a transmit frequency of 350MHz for one path (no cancellation) and for the complete 18-path system (lower part of fig. 8-6). Clearly all problematic products are suppressed significantly. Please note that the

unfortunate FM-radio spurs that are modulated with our output signal are caused by a 100MHz high power FM-radio broadcast transmitter on the roof of our building. Overall, 10 chips were measured with spurious emissions <-40dBc for all harmonics up to the 17th harmonic of the LO, for an LO-frequency from 30-800MHz. For higher frequencies the chip has a 6-path mode which was measured for 30MHz- 2.4GHz with similar rejection up to the fifth harmonic of the LO. The rejection of products related to the fundamental of the LO, like the LO-feedthrough and image component, can be a few dB worse, but requirements on in-band products are usually less strict than for out-of-band spurious emissions.

The (drain) efficiency of the core of the power upconverter is 11%, which is good compared to other power upconverters, given the low harmonics. However, we used current-mode logic circuits biased at high currents at 8GHz LO frequency. As a result the power consumption of the digital part currently dominates (~150mW). In the following sections we examine alternative architectures for multi-phase clock generation, and will look at possibilities to reduce the power consumption while still achieving a low phase error.

8. MULTI-PHASE CLOCK GENERATION REQUIREMENTS

As discussed in the previous sections, polyphase multipath circuits require multi-phase clocks. Such clocks are also useful in many other applications. For quadrature down-conversion mixers, two differential clocks with 90 degrees phase-separation are needed (or four single ended clocks with phases 0, 90, 180 and 270 degrees). A popular harmonic rejection mixer architecture [5,3,8] needs 8 equidistant phases with 45 degree separation. For high-speed serial links multi-phase clocks are used [14] to process data streams at a bit rate higher than the clock frequency, and in time-interleaved ADCs to realize a conversion rate higher than feasible with individual quantizers [15]. Aiming for multi-functionality (e.g. software defined radio), we would like a flexible Multi-Phase Clock Generator (MPCG) to adapt to largely different data rates, sampling rates or radio frequencies.

To implement a MPCG, both delay-locked loops (DLLs) and shift registers (SRs) have been used. A SR MPCG also functions as a divide-by-N divider for N-phase clock generation. Although a SR MPCG seems more attractive due to its wide working frequency range (flexibility), it requires an N times higher clock-frequency and at first glance seems to consume more power. However, a SR MPCG doesn't have jitter accumulation from one clock phase to the other as in a DLL equivalent, which should be taken into

account for a fair comparison. In the following sections, we aim to make a solid comparison between these two MPCGs, primarily based on their power and absolute output jitter performance [16]. Furthermore, flexibility aspects relevant for multi-functionality will be discussed.

We will start with a DLL MPCG, discuss its architecture and analyze its jitter performance, and then addresses the SR MPCG. Later in the chapter we will make a comparison and verify the analysis via simulation results.

9. DLL MPCG JITTER

9.1 DLL MPCG Architecture

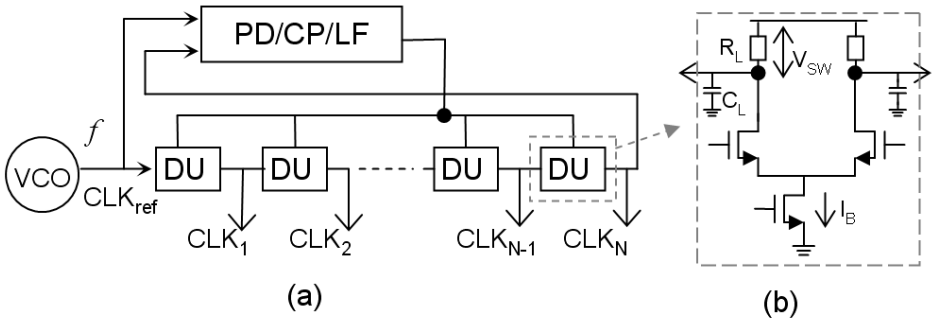

Figure 8-7. (a) DLL MPCG architecture (b) CML delay unit schematic

The architecture of a DLL MPCG is shown in Fig. 8-7(a). It consists of a voltage controlled delay line (VCDL) which has N identical delay units (DUs) and a control loop consisting of a phase detector (PD), a charge pump (CP) and a loop filter (LF). In the DLL, a reference clock CLK_{ref}, generated by a VCO with a frequency of f, is propagated through the VCDL. The loop compares the phase of the last output of the VCDL with CLK_{ref} and controls the VCDL so that its total delay time is one reference clock period. Once locking is achieved, the N outputs CLK_1~CLK_N are multi-phase clocks with $2\pi/N$ phase spacing.

9.2 DLL MPCG Output Jitter

The DLL MPCG output jitter can be divided into three parts: 1) jitter transferred from the reference clock, 2) jitter generated by the VCDL and 3) jitter from the control loop. The jitter of the reference clock is transferred to the DLL outputs with some jitter peaking [17][18]. The DLL cannot decrease reference clock jitter, but jitter peaking can be made very small by choosing a low DLL loop bandwidth [17][18]. For an optimal DLL design, the jitter contribution of the control loop is negligible [17] and hence ignored hereafter. Thus, VCDL jitter is our main worry.

In a DLL MPCG, the VCDL generates two types of jitter: random noise jitter caused by *thermal noise* and deterministic mismatch jitter due to *mismatch* of the delay units. The DLL renders no improvement of VCDL noise jitter. Again, the VCDL noise jitter is lowest for low values of the loop bandwidth, in which case it would be almost equal to that of a free-running VCDL [17]. The jitter will thus accumulate from one delay unit to the other. If the noise jitter variance of one delay unit is $\sigma^2_{t,DU,noise}$, and we assume uncorrelated white noise, the noise jitter variance on the output of the n^{th} delay unit will be n times bigger. For multi-phase clock applications like the software defined radio transmitter discussed in the beginning of this chapter [12], the jitter of every clock phase is equally relevant. To quantify the jitter of a set of N-phase clocks, the averaged jitter variance of the N clocks is a meaningful quantity. The average noise jitter variance generated by the DLL can be calculated as:

$$(\sigma^2_{t,DLL,noise})_{avgN} = \frac{1}{N} \cdot \sum_{n=1}^{N} n \cdot \sigma^2_{t,DU,noise} = \frac{N+1}{2} \sigma^2_{t,DU,noise} \tag{3}$$

Different from noise jitter, the DLL loop *can* improve the deterministic mismatch jitter. The start and end of the VCDL are both aligned to the reference clock and thus have zero deterministic time error. The maximum mismatch jitter appears at the middle of the VCDL. If we define the mismatch jitter variance of one delay unit as $\sigma^2_{t,DU,mis}$, the jitter variance on the output of the n^{th} delay unit can be calculated as [17]:

$$\sigma^2_{t,DU_n,mis} = \frac{n(N-n)}{N} \sigma^2_{t,DU,mis} \tag{4}$$

The average mismatch jitter variance generated is then:

$$(\sigma^2_{t,DLL,mis})_{avgN} = \frac{N^2-1}{6N} \sigma^2_{t,DU,mis} \overset{N^2>>1}{\approx} \frac{N}{6} \sigma^2_{t,DU,mis} \tag{5}$$

10. SR MPCG JITTER

10.1 SR MPCG Architecture

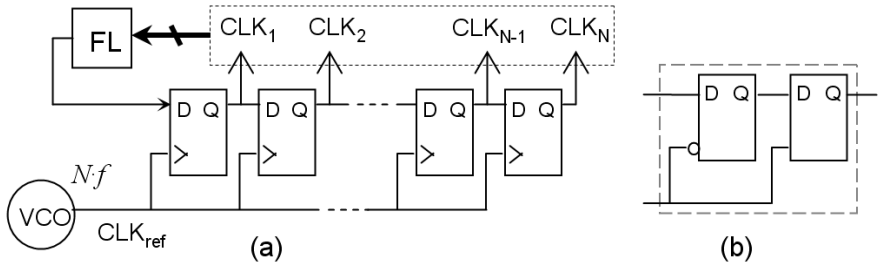

Figure 8-8. (a) SR MPCG architecture (b) DFF block schematic

The architecture of a SR MPCG is shown in Fig. 8-8(a). It consists of a D flip-flop (DFF) chain with N identical DFFs. A reference clock CLK_{ref}, generated by a VCO with a frequency $N{\cdot}f$, is fed into the DFF chain. A flip logic (FL) circuit monitors the N outputs of the DFF chain and flips the logic value at the D input of the first DFF twice every N reference clock cycles. In other words, the outputs of the DFF chain run at a frequency of f and the SR based MPCG also functions as a divide-by-N divider. Since a DFF is sensitive to rising edges, the Q output of each DFF is delayed from the previous DFF's output by one reference clock period, which is equivalently a $2\pi/N$ phase delay. In this way, N-phase clocks $CLK_1 \sim CLK_N$ are generated. Depending on different implementations of the flip logic, the duty cycle of the N-phase clocks can theoretically vary from $1/N$ to $(N\text{-}1)/N$. For example, if 18-phase clocks with a 1/3 duty cycle are wanted, the flip logic can simply be a NOR-gate with CLK_6 and CLK_{12} as its inputs [12]. This gives the SR based MPCG extra flexibility.

10.2 SR MPCG Output Jitter

The SR MPCG output jitter can be divided into two parts: jitter transferred from the reference clock and jitter generated by the DFF chain. The flip logic is simply a logical "enabler" for the first DFF and will not contribute to jitter.

For the jitter transferred from the reference clock, the SR MPCG renders no improvement. Any timing error at the reference clock will be transferred to the DFF chain outputs.

Similar to the VCDL, the DFF chain also generates two types of jitter: noise jitter and mismatch jitter. However, there is *no jitter accumulation* from one DFF to the other, since each DFF output only acts as an "enabler" for the next DFF, while the VCO defines the timing. A DFF can be designed with two master/slave latches as shown in Fig. 8-8(b). For a proper design, only the second latch contributes to jitter since the first is just an "enabler". If we define the rms noise and mismatch jitter variance of one latch as $\sigma^2_{t,Latch,noise}$ and $\sigma^2_{t,Latch,mis}$ respectively, the average jitter variance for the set of N-phase clocks generated by the SR can be easily calculated as:

$$(\sigma^2_{t,SR,noise})_{avgN} = \frac{1}{N} \cdot \sum_{n=1}^{N} \sigma^2_{t,Latch,noise} = \sigma^2_{t,Latch,noise} \tag{6}$$

$$(\sigma^2_{t,SR,mis})_{avgN} = \frac{1}{N} \cdot \sum_{n=1}^{N} \sigma^2_{t,Latch,mis} = \sigma^2_{t,Latch,mis} \tag{7}$$

11. COMPARISON BETWEEN DLL AND SR JITTER

11.1 Comparing Jitter Transferred from the Reference Clock

From the analysis above, we see that both the DLL and SR MPCGs render no improvement on the reference clock jitter. However, the SR MPCG needs a reference clock with N times higher frequency than the DLL. If both clocks are generated by a VCO[1], the VCO for the SR should work at N times higher frequency, raising the question how this impacts power consumption. Assuming the VCO has an f^{-2} power spectrum and its quality of design is adequately assessed via the often used figure of merit *FOM* [19], the single sideband phase noise to carrier ratio at an offset frequency f_m can be expressed as:

[1] The VCO can be part of a synthesizer, e.g., a PLL. We didn't discuss the effect of the PLL loop on the reference clock phase noise since it's the same for the SR and DLL. The PLL for the SR does not require an extra divide-by-N since the SR itself functions as a divide-by-N and can be re-used.

$$L(f_m) = \frac{10^{FOM/10}}{P_{VCO}} \cdot \frac{f_{VCO}^2}{f_m^2} \tag{8}$$

where f_{VCO} is the frequency and P_{VCO} is the power dissipation in [mW]. It is well-known that the variance for stationary absolute jitter is related to the total area of its power spectrum, i.e. the reference clock jitter variance $\sigma^2_{t,ref}$ becomes:

$$\sigma^2_{t,ref} = \frac{2 \times \int_{f_l}^{f_h} L(f_m) d(f_m)}{(2\pi f_{VCO})^2} = \frac{10^{FOM/10}}{2\pi^2 \cdot P_{VCO}} \cdot \left(\frac{1}{f_l} - \frac{1}{f_h}\right) \tag{9}$$

where $[f_l, f_h]$ is the specified integration region. Equation (9) indicates that although the VCO in the SR MPCG runs at N times higher frequency, it outputs the same jitter, given the same power and the same quality of design. If an *LC* VCO is used, higher working frequency may even be preferred, since the quality factor of an inductor ($\omega L/R$) increases with frequency and smaller inductors are needed (less chip area). On the other hand there are limits to increasing the frequency, and also clock buffer power consumption can become an issue.

11.2 Comparing Jitter Generated Due to Thermal Noise

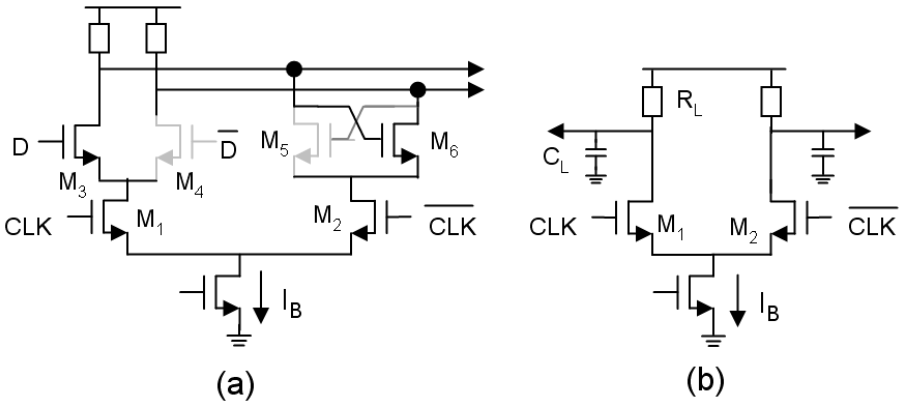

Figure 8-9. (a) Schematic of a CML latch at the switching instant. (b) Simplified schematic for jitter analysis.

Because of better supply noise rejection, current mode logic (CML) circuits are often used in low jitter designs. To compare the jitter generated by the two MPCGs, we assume that they both use CML circuits. The simplified schematic of a CML delay unit is shown in Fig. 8-7(b). It is based on an NMOS source coupled differential pair driving the resistive load R_L and biased by a current source I_B. As the loads are *RC* circuits, the propagation delay t_d can be approximated as:

$$t_d = \ln 2 \cdot R_L C_L = \ln 2 \cdot (V_{SW} / I_B) \cdot C_L \tag{10}$$

where V_{SW} is the differential output swing and is determined by R_L and I_B due to the full switching of the tail current.

The CML implementation of a latch is shown in Fig. 8-9(a). For a proper operation, the *D* inputs of the latch should be already stable before the *CLK* starts to switch. For example, D is high and $\overline{D}$ is low. Therefore, at the switching moment, transistors M4 and M5 are off. M3 and M6 are in their saturation region and work as cascode transistors on top of the differential pair. The noise contribution of M3-M6 can thus be neglected. The schematic of the latch can be simplified to Fig. 8-9(b) which is exactly the same as the schematic of the CML delay unit in Fig. 8-7(b). Therefore, we can apply the same noise jitter analysis for the delay unit and the latch.

The noise jitter variance of a CML delay unit can be predicted using the analysis presented in [20] as:

$$\sigma^2_{t,noise} = (1 + \gamma + \gamma_T \cdot \frac{2I_B}{V_{OV,T}} \cdot \frac{R_L}{2}) \cdot \frac{2kTC_L}{I_B^2} \tag{11}$$

where γ and γ_T are respectively the noise factor of the differential pair transistors and the tail bias transistor, $V_{OV,T}$ is overdrive voltage of the tail bias transistor and $2I_B/V_{OV,T}$ represents its transconductance assuming a square-law model.

In most of the clock generator designs, jitter and power are two important parameters. Via admittance level scaling [21], both noise and mismatch jitter can always be reduced at the cost of increasing the power consumption *P*. In order to take this tradeoff into account and make a fair comparison, jitter variance is normalized to power, with 1mW as reference:

$$(\sigma_t^2)_{NorP} = \sigma_t^2 \cdot (P / 1mW) \tag{12}$$

For a given circuit, applying admittance level scaling will not change the value of $(\sigma^2_t)_{NorP}$. Smaller $(\sigma^2_t)_{NorP}$ means generating less jitter for a given amount of power. For a CML circuit, the power consumption is dominated

by the static power $I_B \cdot V_{DD}$. With Eq. (11) and Eq. (12), we find for both a CML delay unit and latch:

$$(\sigma^2_{t,noise})_{NorP} = (1+\gamma+\gamma_T \cdot \frac{I_B R_L}{V_{OV,T}}) \cdot \frac{2kT \cdot V_{DD}}{1mW} \cdot \frac{C_L}{I_B} \tag{13}$$

Substituting Eq. (10) into Eq. (13) yields:

$$(\sigma^2_{t,noise})_{NorP} = \{(1+\gamma+\gamma_T \cdot \frac{V_{SW}}{V_{OV,T}}) \cdot \frac{2kT \cdot V_{DD}}{\ln 2 V_{SW} \cdot 1mW}\} \times t_d \tag{14}$$

Equation (14) indicates that the *normalized noise jitter variance is proportional to* t_d for a given power budget.

In a DLL, if t_d is tuned by tuning R_L while keep V_{SW} constant, I_B and thus $V_{OV,T}$ in Eq. (14) will vary with t_d. Here to simplify the comparison, we ignore this second order effect and assume the delay unit and the latch have the same V_{SW} and $V_{OV,T}$. We will see the effect of this simplification in Section V. A DLL has N delay units contributing to jitter and power while a SR has N latches contributing to jitter and $2N$ latches dissipating power. The average noise jitter variance generated by the DLL and the SR MPCGs can then be compared using Eqs. (3), (6) and (14), as:

$$\frac{(\sigma^2_{t,SR,noise})_{avgN,NorP}}{(\sigma^2_{t,DLL,noise})_{avgN,NorP}} = \frac{(\sigma^2_{t,Latch,noise})_{NorP} \times 2N}{\frac{N+1}{2} \times (\sigma^2_{t,DU,noise})_{NorP} \times N} = \frac{4}{N+1} \cdot \frac{t_{d,Latch}}{t_{d,DU}} \tag{15}$$

The comparison result thus depends on the amount of delay of the delay unit $t_{d,DU}$ and that of the latch $t_{d,Latch}$. In a DLL MPCG, the VCO defines the frequency and the VCDL defines the delay in between the N output clocks. Both the VCO and the delay line need to be tuned for the DLL MPCG to work at a frequency f, where the delay of each delay unit should satisfy:

$$t_{d,DU} = \frac{T}{N} = \frac{1}{N \cdot f} \tag{16}$$

In contrast, the SR MPCG is more flexible. For different f, only the VCO needs to be tuned since both the frequency and the delay in between the N output clocks are defined by the clock period of the VCO. The only concern is that the DFFs should operate correctly, which requires [22]:

$$t_{d,Latch} + t_{su} \le \frac{1}{N \cdot f} \tag{17}$$

where t_{su} is the setup time required by the DFF. Defining the maximum working frequency of a SR MPCG for N-phase clock generation in a certain technology as $f_{max,SR}$, the latch delay will have its minimum value $t_{d,Latch,min}$ at $f_{max,SR}$ given by:

$$t_{d,Latch,\min} = \frac{1}{1+\alpha_{su}} \cdot \frac{1}{N \cdot f_{\max,SR}} \tag{18}$$

with α_{su} the ratio between t_{su} and $t_{d,Latch,min}$. As a small delay is preferred for a small $(\sigma^2_{t,noise})_{NorP}$, the latch delay can be equal to its minimum in Eq. (18). For a delay unit, the delay is limited by Eq. (16). Taking this factor into account, Eq. (15) can be re-written as:

$$\frac{(\sigma^2_{t,noise,SR})_{avgN,NorP}}{(\sigma^2_{t,noise,DLL})_{avgN,NorP}} = \frac{1}{1+\alpha_{su}} \cdot \frac{f}{f_{\max,SR}} \cdot \frac{4}{N+1} \tag{19}$$

As soon as the wanted number of clock phases is larger than three (N>3), Eq. (19) is smaller than one since the DFF needs a finite setup time (α_{su}>0) and the working frequency of the SR can't surpass the technology limit ($f \le f_{max,SR}$). This means that the SR based MPCG generates less noise jitter than the DLL counterpart for a given power budget. Equation (19) also indicates that the advantage of the SR based MPCG will be larger if more advanced technologies are used and in applications where clocks with a larger number of phases at lower frequencies are needed.

11.3 Comparing Jitter Generated Due to Mismatch

Based on similar reasoning as for the noise jitter analysis, the latch can be simplified as shown in Fig. 8-9(b) for mismatch jitter analysis and we can apply a similar analysis. In a CML delay unit, there are two mismatch jitter sources: one is the RC load which contributes to RC delay mismatch $\sigma^2_{t,RC,mis}$ and the other is the differential pair input referred offset voltage σ^2_{Voff} which makes the switching moment deviate from the actual crossing point of the input clocks. The tail bias transistor mismatch does not lead to jitter since it's a common mode error and we are interested in the crossing points.

Using Eq. (10), the jitter due to the RC load mismatch becomes:

$$(\frac{\sigma_{t,RC,mis}}{t_d})^2 = \sigma^2_{\Delta R_L/R_L} + \sigma^2_{\Delta C_L/C_L} \tag{20}$$

with ΔR_L and ΔC_L the absolute error in the value of R_L and C_L.

In a DLL, the *RC* delay must be tunable. For simplicity, we assume that C_L is tuned by putting less or more capacitors in parallel and R_L is tuned by putting less or more resistors in parallel[2]. Since the matching improves with area [21], Eq. (20) can be rewritten as:

$$\sigma^2_{t,RC,mis} = [(A_R \cdot \sqrt{R_L})^2 + (A_C / \sqrt{C_L})^2] \times t_d^2 \tag{21}$$

where A_R and A_C are IC process constants for the matching property of the load resistance and capacitance, respectively.

The input referred offset voltage of a differential pair can be calculated using the method presented in [23] as:

$$\sigma^2_{Voff} = \sigma^2_{\Delta Vt} + \frac{I_B}{4K} \times \sigma^2_{\Delta R_L'/R_L} + \frac{I_B}{4K} \times \sigma^2_{\Delta K/K} \tag{22}$$

where $\sigma^2_{\Delta Vt}$ is the differential pair threshold voltage mismatch variance, $\Delta R'_L$ is the relative error between the two R_L loads, K is the transconductance parameter of the differential pair with $\sigma^2_{\Delta K/K}$ describing its mismatch.

The total mismatch jitter variance $\sigma^2_{t,mis}$ can be found by adding $\sigma^2_{t,RC,mis}$ and the jitter variance caused by σ^2_{Voff} which is σ^2_{Voff} divided by $(I_B/C_L)^2$, the square of the slope of the differential switching voltage at the zero crossing.

$$\sigma^2_{t,mis} = A_R^2 \cdot R_L \cdot t_d^2 + \frac{A_C^2 \cdot t_d^2}{C_L} + \frac{\sigma^2_{\Delta Vt} + \frac{I_B}{4K} \times A_R^2 \cdot R_L + \frac{I_B}{4K} \times \sigma^2_{\Delta\beta/\beta}}{(I_B/C_L)^2} \tag{23}$$

The power normalized mismatch jitter variance can be derived with Eq. (12) and Eq. (23) as:

$$\begin{aligned}(\sigma^2_{t,mis})_{NorP} = \frac{V_{DD}}{1mW} \cdot \{ & V_{SW} \cdot A_R^2 \times t_d^2 + \ln 2 \cdot V_{SW} \cdot A_C^2 \times t_d \\ & + \frac{\sigma^2_{\Delta Vt}}{\ln 2 \cdot V_{SW}} \times C_L \cdot t_d + \frac{A_R^2}{\ln 2 \times 4K} \times C_L \cdot t_d + \frac{\sigma^2_{\Delta K/K}}{4K} \times C_L^2 \}\end{aligned} \tag{24}$$

[2] If R_L is realized with a MOS transistor in linear region and tuned by tuning the gate voltage, it can be shown that the matching property of R_L in a DLL DU is even worse.

Equation (24) shows that the delay unit and latch generates less mismatch jitter for a smaller delay, with a given power. It also suggests that with a constant V_{SW}, it's better for a DLL to tune up R_L instead of C_L when larger delay is needed.

Assuming the terms with t_d proportionality in Eq. (24) which include the threshold voltage mismatch are the dominating mismatch jitter sources and setting the other initial conditions the same for a fair comparison, the mismatch jitter generated by the DLL and SR can be compared with Eqs. (5), (7) and (24) as:

$$\frac{(\sigma^2_{t,SR,mis})_{avgN,NorP}}{(\sigma^2_{t,DLL,mis})_{avgN,NorP}} \approx \frac{12}{N} \cdot \frac{t_{d,Latch}}{t_{d,DU}} \tag{25}$$

Substituting Eq. (16) and Eq. (18) into Eq. (25) yields:

$$\frac{(\sigma^2_{t,SR,mis})_{avgN,NorP}}{(\sigma^2_{t,DLL,mis})_{avgN,NorP}} = \frac{1}{1+\alpha_{su}} \cdot \frac{f}{f_{\max,SR}} \cdot \frac{12}{N} \tag{26}$$

The situation where Eq. (26) is larger than one only occurs when the wanted number of clock phases N is smaller than 12 together with a high frequency f close to $f_{max,SR}$. In other cases, Eq. (26) is smaller than one, which means that the SR MPCG generates less mismatch jitter than the DLL counterpart for a given power budget. Equation (26) also indicates that the advantage of the SR based MPCG will be larger if more advanced technologies are used and a larger number of clock phases at lower frequencies are needed.

11.4 Discussion

The analysis above shows that a SR MPCG transfers the same jitter from the reference clock and almost always generates less jitter[3] than a DLL MPCG for a given power consumption. For mismatch jitter, the DLL MPCG may have a slight advantage in some high frequency cases[4]. Although we assumed that current mode logic circuits are used to implement the MPCG, the way of analysis developed can also be applied when other logic families like CMOS logic, true single phase clocking or dynamic transmission gate

[3] In case phase noise is important, the SR is also better as both the SR and DLL generate white phase noise, while the reference clock has the same spectrum shape for both cases.

[4] If 50% reference clock duty cycle is guaranteed, both edges can be used. The N DFFs in the SR can be replaced with N latches as in [12]. The previous analysis then overestimates the SR MPCG power consumption by two times.

logic are used. Note that the advantage of a SR MPCG comes from its features like no jitter accumulation from one clock phase to the other and the flexibility of setting small latch delay time. These features are independent of the logic family used.

From an implementation point of view, the SR MPCG is easier since it does not require a phase detector, loop filter and analog tuned delays. However, it can be difficult to implement in applications where N is large and f is high since the SR works at $N{\cdot}f$. Still, speed improves as technology advances. Another concern is that the loading of the VCO is more severe in the SR MPCG, since it needs to drive N DFFs. This problem can be alleviated by downscaling the DFFs by admittance scaling [21], which is acceptable because they generate less jitter than the delay units, thus saving power and chip area.

From a multi-functionality point of view, the SR MPCG is clearly more attractive: it is basically a digital circuit which can operate from arbitrarily low frequency up to $f_{max,SR}$, while a DLL requires tuning of an "analog" delay. Also, a SR can basically instantaneously change its output frequency, while a DLL settles slowly, due to the preferred low loop bandwidth. Finally, a SR MPCG has the flexibility to generate clocks with different duty cycle.

12. SIMULATION RESULTS

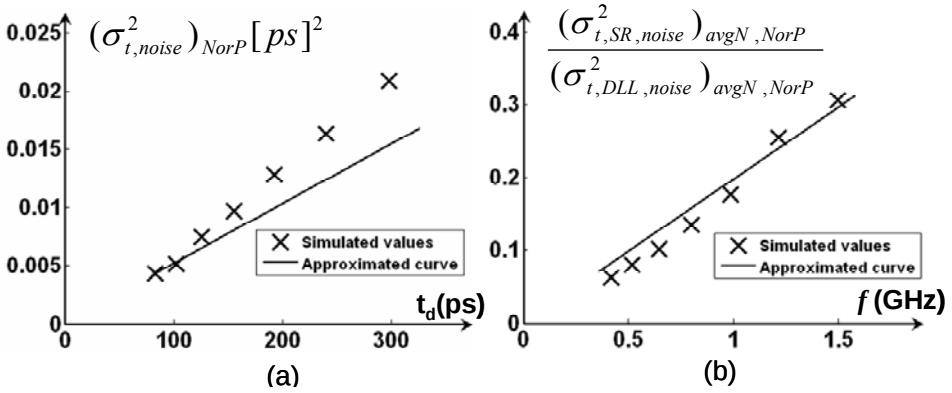

Figure 8-10. Noise jitter simulation results in 0.13μm CMOS with N=8 for (a) a CML delay unit (b) DLL and SR comparison.

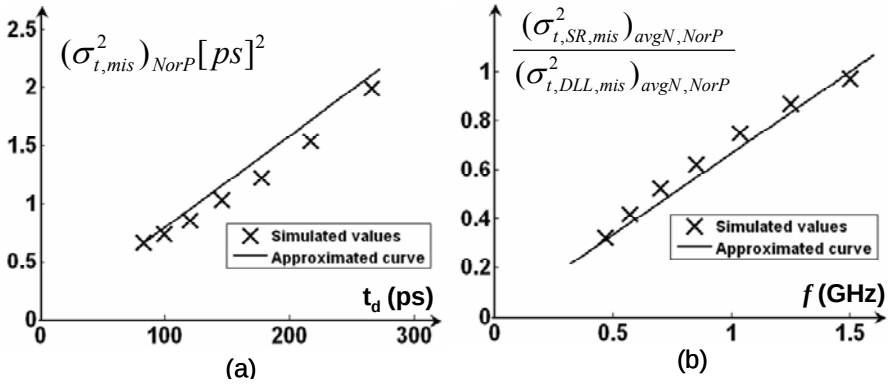

Figure 8-11. Mismatch jitter simulation results in 0.13µm CMOS with N=8 for (a) a CML delay unit (b) DLL and SR comparison.

In order to verify the calculations, simulations were done for a DLL and a SR for N=8 in 0.13-μm CMOS. The reference clocks are voltage sources with 1kohm source resistance. The VCDL delay is tuned up by tuning up the load resistance as suggested by Eq. (24) while keep V_{SW} to be 0.6V. For the DFFs, α_{su} is about 0.5. The load capacitance is 100fF, which is comparable to the parasitic capacitances. In this implementation, $f_{max,SR}$ is about 1.5 GHz for 8-phase clock generation. Fig. 8-10 shows the strobed PNoise analysis results for noise jitter. The simulated values coarsely fit the estimated curve. The larger deviation when t_d is larger relates to the simplification we made below Eq. (14). We see this simplification is in favor of the DLL which normally has a larger t_d. Therefore, it does not affect the conclusion. Fig. 8-11 shows the Monte Carlo analysis results for mismatch jitter. The bent shape of the simulated values when t_d is tuned from low to high is predicted by Eq. (24). The simulated values fit the estimated curve well which means the threshold voltage mismatch dominates in this design.

13. CONCLUSION

In this chapter we reviewed some recent research results relevant for the feasibility of fully integrated CMOS cognitive radio transceivers. We motivated why an ADC and DAC are not sufficient to realize the radio interface. Coarse power estimates show that A/D conversion of high dynamic range radio signals at the antenna is not realistic for GHz radio signal. However, RF sampling is feasible and the sampling clock jitter requirements are not as difficult as often thought, but are similar to those of traditional mixer based RF receivers. A key fundamental problem in radio circuits is their nonlinear and/or time-variant nature. As a result they produce

not only a wanted output signal, but also many unwanted harmonics and sidebands. We presented a polyphase multipath technique that addresses this problem without using any dedicated filters. Using this technique, a highly flexible power up-converter has been realized on in CMOS, operating at an arbitrary transmit frequency between DC and 2.4GHz, with unwanted harmonics and sideband lower than <-40dBc.

Flexible multi-phase clock generation is at the heart of multipath polyphase transceivers. This chapter motivates why a SR MPCG is more attractive for flexible multi-functional circuits than a DLL MPCG as it is easier to change its frequency and duty cycle. Furthermore, analysis shows that a SR MPCG almost always generates less jitter than a DLL equivalent when both are realized with CML circuits, at a given power budget. This is partly because a SR MPCG has no jitter accumulation from one clock phase to the other as in a DLL counterpart. In addition, a SR MPCG can use latches with very small delay time, while jitter generation of a CML circuit is proportional to its (functionally required) delay time. A SR MPCG requires a reference clock with higher frequency, which can be realized in a power neutral way provided that the VCO core determines power consumption. The advantages of a SR MPCG will be larger as technology advances.

ACKNOWLEDGEMENTS

The authors would like to thank Eisse Mensink and Rameswor Shresta for contributions to this work, Henk de Vries and Gerard Wienk for practical assistance during design and measurements and Fokke Hoeksema and Jaap Haartsen for useful discussions. Philips Research is acknowledged for providing the silicon.

REFERENCES

[1] M. J. Marcus, "Unlicensed cognitive sharing of TV spectrum: the controversy at the Federal Communications Commission," *IEEE Communications Magazine*, vol. 43, pp. 24-25, 2005.

[2] R. H. Walden, "Performance trends for analog to digital converters," *IEEE Communications Magazine*, vol. 37, pp. 96-101, 1999.

[3] R. Bagheri, A. Mirzaei, M. E. Heidari, S. Chehrazi, L. Minjae, M. Mikhemar, W. K. Tang, and A. A. Abidi, "Software-defined radio receiver: dream to reality," *IEEE Communications Magazine*, vol. 44, pp. 111-118, 2006.

[4] F. Bruccoleri, E. A. M. Klumperink, and B. Nauta, "Wide-band CMOS low-noise amplifier exploiting thermal noise canceling," *IEEE J. Solid-State Circuits*, vol. 39, pp. 275-282, 2004.

[5] J. A. Weldon, R. S. Narayanaswami, J. C. Rudell, L. Li, M. Otsuka, S. Dedieu, T. Luns, T. King-Chun, L. Cheol-Woong, and P. R. Gray, "A 1.75-GHz highly integrated narrow-band CMOS transmitter with harmonic-rejection mixers," *IEEE J. Solid-State Circuits*, vol. 36, pp. 2003-2015, 2001.

[6] K. Muhammad, R. B. Staszewski, and D. Leipold, "Digital RF processing: toward low-cost reconfigurable radios," *IEEE Communications Magazine*, vol. 43, pp. 105-113, 2005.

[7] V. J. Arkesteijn, E. A. M. Klumperink, and B. Nauta, "Jitter requirements of the sampling clock in software radio receivers," *IEEE Transactions on Circuits and Systems II: Express Briefs,* vol. 53, pp. 90-94, 2006.

[8] Z. Ru, E.A.M. Klumperink, B. Nauta, "A Discrete-Time Mixing Receiver Architecture with Wideband Harmonic Rejection", *2008 IEEE International Solid-State Circuits Conference (ISSCC), Digest of Technical Papers*, pp. 322-323+616, 2008.

[9] W. Sansen, "Distortion in elementary transistor circuits," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on]*, vol. 46, pp. 315-325, 1999.

[10] P. B. Kenington, "Linearized transmitters: an enabling technology for software defined radio," *IEEE Communications Magazine*, vol. 40, pp. 156-162, 2002.

[11] E. Mensink, E. A. M. Klumperink, and B. Nauta, "Distortion cancellation by polyphase multipath circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 52, pp. 1785-1794, 2005.

[12] R. Shrestha, E. Mensink, E. A. M. Klumperink, G. J. M. Wienk, and B. Nauta, "A Polyphase Multipath Technique for Software Defined Radio Transmitters", *IEEE Journal of Solid State Circuits*, Vol. 41, No. 12, pp. 2681-2692, Dec. 2006.

[13] E. A. M. Klumperink, S. M. Louwsma, G. J. M. Wienk, and B. Nauta, "A CMOS switched transconductor mixer," *IEEE Journal of Solid-State Circuits*, vol. 39, pp. 1231-1240, 2004.

[14] C. K.Yang and M. A.Horowitz, "A 0.8-/spl mu/m CMOS 2.5 Gb/s oversampling receiver and transmitter for serial links," *IEEE J. Solid-State Circuits*, vol. 31, pp. 2015-2023, Dec. 1996.

[15] W. C. Black, and D. A. Hodges, "Time interleaved converter arrays", *IEEE J. Solid-State Circuits*, vol.15, no. 6, pp. 1022–1029, Dec. 1980.

[16] X. Gao, E. Klumperink and B. Nauta, "Advantages of Shift Registers Over DLLs for Flexible Low Jitter Multiphase Clock Generation", *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 55, no.3, pp. 244 -248, Mar. 2008.

[17] R. van de Beek, E. Klumperink, C. Vaucher, and B. Nauta, "Low-jitter clock multiplication: a comparison between PLLs and DLLs," *IEEE Trans. Circuits Syst. II*, vol. 49, pp. 555-566, Aug. 2002.

[18] M.-J. Edward Lee, et al., "Jitter transfer characteristics of delay-locked loops-theories and design techniques," *IEEE J. Solid-State Circuits*, vol. 38, pp. 614-621, Apr. 2003.

[19] P. Kinget, "Integrated GHz voltage controlled oscillators," *Analog Circuit Design: (X)DSL and Other Communication Systems; RF MOST Models; Integrated Filters and Oscillators*, W.Sansen, J.Huijsing, and R.van de Plassche, Ed. Boston, MA: Kluwer, 1999, pp. 353-381.

[20] S. Levantino, et al., "Phase noise in digital frequency dividers", *IEEE J. Solid-State Circuits*, vol. 39, no.5, pp. 775 – 784, May 2004.

[21] E. Klumperink, B. Nauta, "Systematic Comparison of HF CMOS Transconductors", *IEEE Trans. Circuits Syst. II*, vol. 50, no.10, pp. 728 -741, Oct. 2003.

[22] J. M. Rabaey, *Digital Integrated Circuits, A Design Perspective* Englewood Cliffs, NJ: Prentice-Hall, 1996.

[23] P. Gray, P. Hurst, S. Lewis, and R. Meyer, *Analysis and Design of Analog Integrated Circuits*, 4th Edition. John Wiley & Sons, Inc., 2001, pp.236-237.

[24] E.A.M. Klumperink, R. Shrestha, E. Mensink, V.J. Arkesteijn, B. Nauta, "Cognitive radios for dynamic spectrum access - Polyphase Multipath Radio Circuits for Dynamic Spectrum Access", *IEEE Communications Magazine*, Volume 45, Issue 5, pp. 104 – 112, May 2007.