A 200 μA Duty-Cycled PLL for Wireless Sensor Nodes

Salvatore Drago*, Domine Leenaerts*, Bram Nauta†, Fabio Sebastiano*, Kofi Makinwa‡ and Lucien Breems*
*NXP Semiconductors, Eindhoven, The Netherlands, Email: salvatore.drago@nxp.com
‡Electronic Instrumentation Laboratory, Delft University of Technology, Delft, The Netherlands
†IC Design Group, CTIT Research Institute, University of Twente, Enschede, The Netherlands

***Abstract*— A duty-cycled PLL operating in burst mode is presented. It is an essential building block of a moderately accurate low-power frequency synthesizer suitable for use in nodes for Wireless Sensor Networks. Once in lock, the PLL's frequency error is less than 0.1% (rms). Fabricated in a baseline 65 nm CMOS process, the PLL occupies 0.19X0.15 mm^2 and draws 200 μA from a 1.3-V supply when generating a 1 GHz signal with a duty cycle of 10%.**

I. Introduction

High frequency synthesizers are essential blocks of nearly all analog, digital, and radio-frequency systems. They are often the most power hungry blocks in nodes for Wireless Sensor Networks (WSN) [1], [2]. However, they also need to be accurate, and considerable effort has been devoted to the realization of low-power PLLs with sufficient accuracy [3].

Conventional PLLs can achieve an accuracy of a few ppm, but are usually designed to meet stringent phase noise and spectral purity requirements [3], [4]. These lead to high power consumption, which is not suitable for use in WSN nodes. To reduce power consumption, special architectures with relaxed phase noise and accuracy specifications can be used. In [1], [2] a free-running digital controlled oscillator (DCO) is periodically calibrated. This approach is extremely low power, but its accuracy is limited to only a few percent. Moreover, temperature and voltage supply variations give rise to intolerably large frequency drift.

In WSN nodes, which are characterized by low activity, PLLs can be duty-cycled to save power. This suggests that the PLL be operated in burst mode, in which short bursts of generated signals are separated by long idle periods in which energy is saved. Although burst mode PLLs are less accurate than conventional PLLs, they can achieve accuracies of 0.1%, which easily meet the requirements of WSN applications [5]. Moreover, a PLL, because of its closed loop nature is less prone to frequency drift than a free running oscillator. However, burst-mode PLLs require special architectures to ensure stability and fast start-up circuitry to avoid extra power consumption during the transitions between active and idle periods.

The objective of this work is to present a novel frequency synthesizer able to operate in burst mode while maintaining a frequency error of less than 0.1%. The proposed PLL can be operated at low duty-cycles, since it employs a fast start-up DCO, resulting in a highly energy-efficient synthesizer. The proposed PLL can generate frequencies that range from several hundreds of MHz to more than 1 GHz, while maintaining a power consumption of a few hundreds μW.

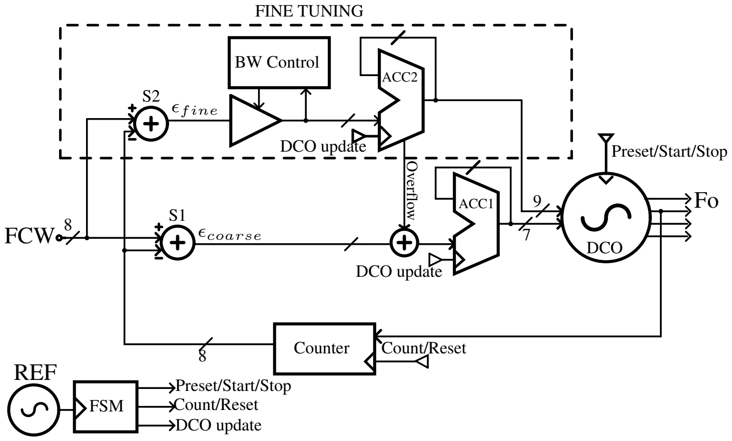

Fig. 1. Duty-cycled PLL.

II. Circuit description

A. Duty Cycled PLL architecture

In order to enable burst mode operation, an All-Digital PLL is preferred to a conventional analog PLL based on a phase frequency detector and a charge pump. This is because the DCO's digital control word (DCW) can be stored in a memory, allowing frequency tracking between two successive bursts. In ultra low power implementations, the oscillator's phase noise is the dominant source of error in both analog and digital PLLs.

A simplified block diagram of the proposed Duty-Cycled PLL (DCPLL) is shown in Fig. 1. Its main loop consists of a DCO, a counter, an accumulator (ACC1) and one digital subtractor (S1). A second fine tuning loop increases the accuracy of the output frequency. Both loops are controlled in an efficient manner by a finite state machine (FSM). The DCO consists of a current-controlled ring oscillator and a 16-bit digital-to-analog converter (DAC) segmented in two banks: one 7-bit bank for coarse frequency acquisition and one 9-bit bank for fine tuning [4].

A reference clock with a frequency REF=20 MHz drives the FSM, which generates the control signals whose timing diagram is shown in Fig. 2. The DCO is periodically turned

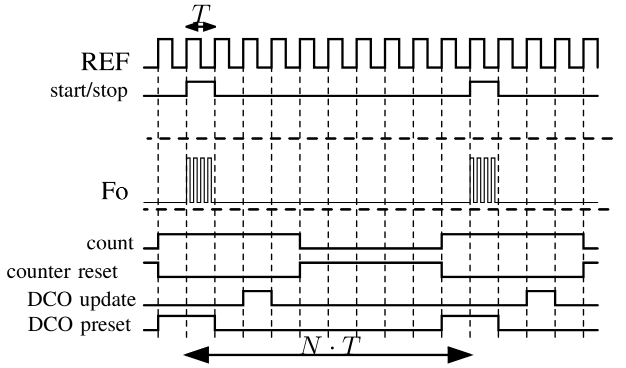

Fig. 2. DCPLL waveforms.

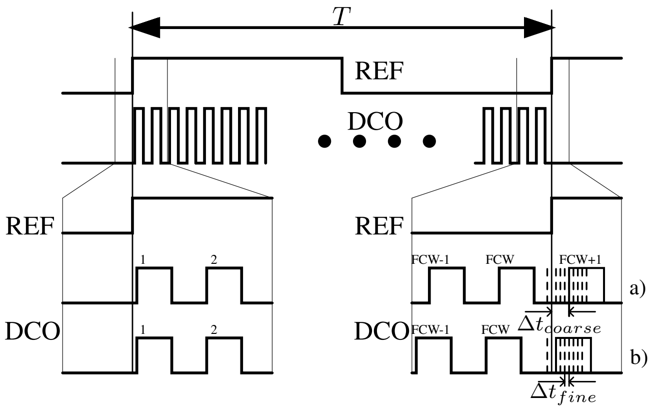

Fig. 3. a) Coarse acquisition b) Fine tuning

on and off, while the two loops ensure that its frequency is locked to REF. After a sleep time of $N-1$ reference clock cycles, the DCO is started up and allowed to run for only one reference clock cycle, i.e. for a period $T = 50$ ns. The DCO drives a clocked counter, which counts the number of DCO rising edges that occur during the reference clock cycle. The resulting integer is compared with the desired frequency control word (FCW) and the resulting error signal updates the DCW stored in the accumulator. Since the error is computed one reference clock after the DCO is stopped, the counter can be implemented as a simple asynchronous D-FF-based counter.

As will be explained in the next section, a short preset period is used to speed-up the DCO's start-up. When locked, the number of DCO rising edges between two reference edges is equal to the programmable FCW. The DCO has a duty-cycle of $1/N$ and its output frequency F_O is:

$$F_O = FCW \cdot REF \tag{1}$$

Conceptually a single loop should be sufficient. However, as shown in Fig. 3 a), the $(FCW + 1)^{th}$ DCO rising edge might be delayed by Δt_{coarse} with respect to the reference rising edge, which results in a worst-case absolute frequency error equal to REF. Significantly better performance can be achieved if, in conjunction with the main loop, which handles the coarse frequency acquisition, an additional loop is employed for fine frequency tuning. A small increase Δf_{fine} of the DCO's frequency advances the last DCO edge by a time interval Δt_{fine} given by:

$$\Delta t_{fine} \simeq \frac{\Delta f_{fine}}{REF} T_{DCO} \tag{2}$$

where T_{DCO} is a DCO period. Before each burst generation, the fine tuning loop increases the DCW by a least significant bit (LSB) increasing the DCO frequency by a small step Δf_{fine} until the $(FCW + 1)^{th}$ DCO edge just leads the reference clock edge. At this point, the fine tuning loop increases or decreases the DCW by 1 LSB depending on whether the $(FCW + 1)^{th}$ DCO edge leads or lags the reference clock edge. Burst by burst, the frequency then varies by $\pm\Delta f_{fine}$ and so the last DCO edge jumps backward and forward around the reference clock edge. While the main loop controls the number of rising edges occurred between two successive reference clock edges, the fine tuning loop decreases the delay between the last DCO rising edge and the reference clock edge. The total error is reduced and the accuracy is improved (Fig. 3 b)).

The quantization error in the frequency generated by the dual loop configuration is proportional to Δf_{fine}. This error can be minimized by increasing the DCO's resolution. However, in a low power implementation, the accuracy is limited by the DCO's phase noise and, hence, by the total power available. In the current design, Δf_{fine} has been chosen low enough to make the quantization noise negligible with respect to the phase noise.

Since the PLL operates in burst mode, the fine tuning operation does not require a power hungry bang-bang phase detector but only requires simple logic circuits [6]. Fig. 4 shows the combined transfer characteristic of the counter and subtractor for the coarse acquisition and fine tuning loops. In the transfer characteristic of the coarse acquisition loop a horizontal dead-band is introduced to produce a null error signal when the integer number of DCO edges falling into one clock cycle is equal to the programmed FCW. In order to realize the bang-bang operation of the fine tuning loop, a vertical dead-band is implemented in its transfer characteristic.

Finally, the fine tuning dynamics are adjusted based on whether the system is in the acquisition or in the steady-state tracking mode. In doing so, both a faster PLL settling time

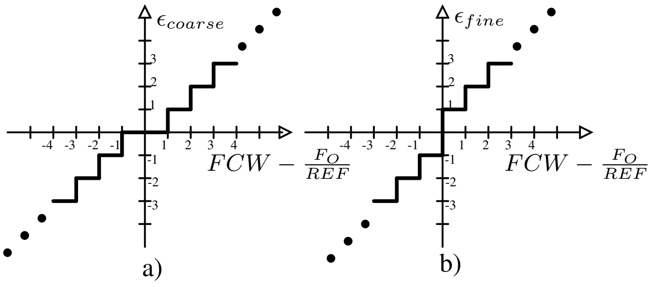

Fig. 4. Transfer characteristic of counter and subtractor: a) Coarse acquisition b) Fine tuning

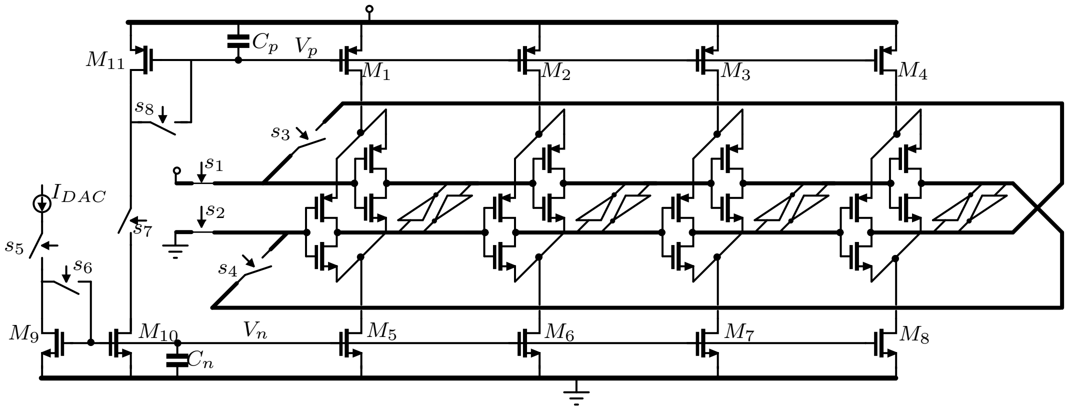

Fig. 5. Schematic of the DCO

and an accurate frequency output can be achieved. By means of the bandwidth control block, the gain in the fine loop can be modified to achieve an adaptive bandwidth.

B. DCO

The proposed DCPLL can work only with a fast start-up DCO whose output frequency can settle within T=50 ns. Ring oscillators start up faster than LC oscillators, which require approximately Q periods, where Q is the quality factor of the LC tank, to reach steady-state [7]. Additionally, if phase noise is not the main requirement, ring oscillators require less power than LC oscillators [2]. Finally, since the DCO will be turned off for a significant fraction of time, its static power consumption in idle mode should be very low. These considerations motivate the use of the ring oscillator shown in Fig. 5. It consists of four delay stages in a closed loop and an R/2R ladder current DAC. Each delay stage uses a pseudo-differential architecture. The frequency is controlled by the complementary voltages V_p and V_n at the gates of PMOS M_1 − M_4 and NMOS M_5 − M_8 which are stored on the two large gate capacitors C_p and C_n. During the idle state, the switches s_1 and s_2 are connected to Vdd and Ground, respectively, while the final stage of the delay line is disconnected from the first stage by means of the switches s_3 and s_4. Therefore, the oscillator's power consumption is only determined by the leakage currents of the inverters. Opening s_1 and s_2 and closing s_3 and s_4 configures the delay line as an oscillator whose output frequency depends on the control voltages V_p and V_n. Most of its power dissipation is due to switching events (i.e. is proportional to CV^2). To synthesize the desired frequency, the per-stage delay is tuned to 1/8 of the desired RF cycle period by means of the DAC current source I_{DAC} which sets the two voltages V_p and V_n. The DCO start-up delay must be negligible with respect to the reference period. This requires that C_p and C_n be large capacitors and that the currents through the diodes M_9 and M_{11} be large enough to set the voltages in a short time. To achieve this while maintaining a low power consumption, a preset phase precedes the DCO's actual start-up. During the preset phase, which begins one reference clock before the DCO is started (Fig. 2), the switches $\mathrm{s}_5 - \mathrm{s}_8$ are enabled and the generated current I_{DAC} sets the voltage V_p and V_n . So when the DCO is started, all voltages are already preset to their correct values, thus mitigating output frequency variations. The DCO is kept running for one reference cycle and, then, shut down by means of the switches $\mathrm{s}_1 - \mathrm{s}_4$ which configure the DCO again as an open-loop delay line. The switches $\mathrm{s}_5 - \mathrm{s}_8$ are opened to preserve the charge in the capacitors C_p and C_n and the DAC is turned off to save power.

III. Experimental results

The oscillator has been realized in a baseline TSMC 65-nm CMOS process. The circuit measures 0.03 mm^2. Most of the area is occupied by the R/2R network and by the two digital loops (Fig. 6). The output frequency can be programmed from 200 MHz to 1.2 GHz. As shown in Fig. 7, the DCPLL's output consists of a train of 1 GHz bursts with a 10% duty-cycle (N=10). The total current consumption at 1.3-V supply voltage is 200 μA (100 μA for the DCO; 60 μA for the current DAC; 40 μA for the counter and PLL logic). The PLL's initial settling transient is shown in Fig. 8. Each point represents the average frequency measured within each burst. After 15 bursts, or equivalently, after 7.5 μs, the output frequency settles to the programmed frequency with an error of 0.1%. In the case shown, the DCO's initial frequency was set to about 300 MHz by loading an estimated DCW into the accumulator. After the PLL's first settling transient, the correct DCW will be

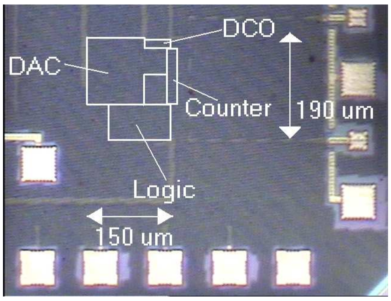

Fig. 6. Die micrograph of the test chip.

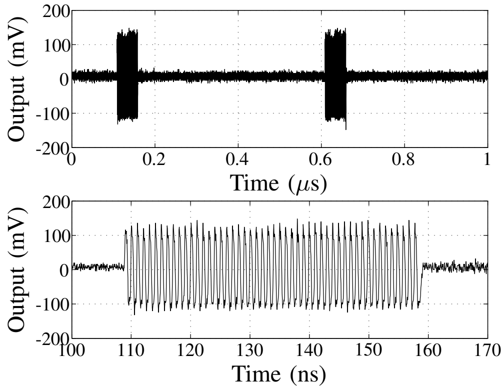

Fig. 7. Measured DCPLL output (top) with zoom-in (bottom).

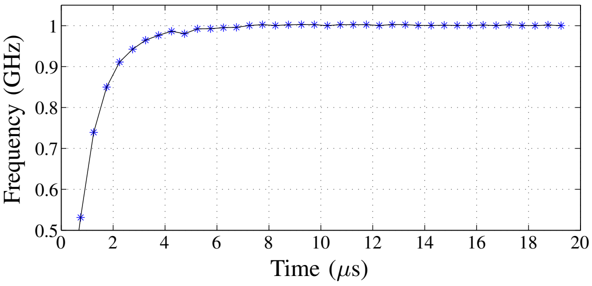

Fig. 8. Measured DCPLL settling time.

stored in the two accumulators and only needs to be adjusted slightly to compensate for temperature and voltage variations. Fig. 9 shows the frequency deviation from the programmed frequency (1 GHz in the present case or FCW=50) for 40 consecutive bursts after one process calibration. Each point represents the average frequency within each burst, while the two bold lines represent the 1σ error. The absence of the systematic "bang-bang" frequency jumps confirms that the accuracy in the generated frequency is limited by the DCO phase noise. The simulated open loop DCO phase noise at 1 MHz offset is, indeed, -73 dBc/Hz, which can be proven to correspond to a DCO period jitter of 7 ps (rms) [8]. After 50 DCO periods, the accumulated jitter for the $(FCW+1)^{th}$ edge is 50 ps giving a time uncertainty of 0.1% with respect to the reference clock. This is in accordance with the measured 0.1% frequency error observed in Fig. 9. It can be seen that the fine tuning loop significantly improves the achieved accuracy; an error of 20 MHz (2%) would be obtained with only the main loop. The standard deviation of the frequency error represents an important parameter for burst-mode frequency synthesizer since it replaces the closed-loop PLL phase noise. To characterize the DCO's performance, its instantaneous DCO frequency during a burst has been measured and is reported in Fig. 10 together with the interpolated frequency (2 samples averaging) and the average frequency over a burst period. The DCO starts approximately at the correct frequency and takes a few DCO periods to settle. The DCPLL is not sensitive to this systematic variations but it tries to tune the average frequency showed as dashed line. However, the deviation from the fixed frequency is kept within few percent thanks to the preset strategy.

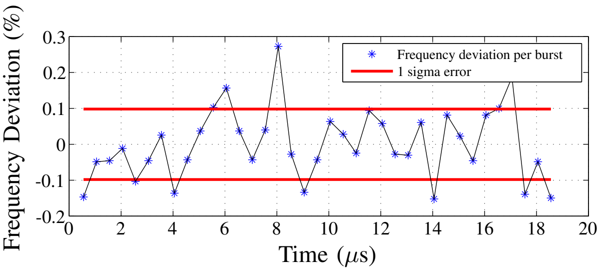

Fig. 9. Measured DCPLL Frequency deviation vs. time

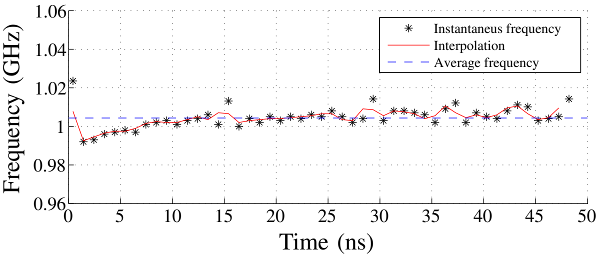

Fig. 10. Measured DCO instantaneous frequency during a burst.

IV. Conclusions

A fully integrated duty-cycled PLL has been presented. Frequency multiplication inaccuracy, due to noise, amounts to 0.1% (1σ). It was shown that a PLL operating in burst mode can be used to generate an high frequency signal accurate enough for WSN applications while maintaining a low power, as required by energy autonomous sensor nodes.

Acknowledgment

This work is funded by the European Commission in the Marie Curie project TRANDSSAT - 2005-020461.

References

[1] B. W. Cook, A. D. Berny, A. Molnar, S. Lanzisera, and K. S. J. Pister, "An ultra-low power 2.4GHz RF transceiver for wireless sensor networks in 0.13μm CMOS with 400mV supply and an integrated passive RX front-end," in *ISSCC, Dig. of Tech. Papers*, Aug. 2006, pp. 258 – 259.

[2] N. Pletcher, S. Gambini, and J. Rabaey, "A 52 μW Wake-Up Receiver With $-$ 72 dBm Sensitivity Using an Uncertain-IF Architecture," *Solid-State Circuits, IEEE Journal of*, vol. 44, no. 1, pp. 269–280, Jan. 2009.

[3] X. Gao, E. A. M. Klumperink, M. Bohsali, and B. Nauta, "A 2.2GHz 7.6mW Sub-Sampling PLL with -126 dBc/Hz In-Band Phase Noise and 0.15 ps Jitter in 0.18μm CMOS," in *ISSCC, Dig. of Tech. Papers*, Feb. 2009, pp. 392–393.

[4] R. Staszewski, *et al.*, "All-digital PLL and transmitter for mobile phones," *Solid-State Circuits, IEEE Journal of*, vol. 40, no. 12, pp. 2469–2482, Dec. 2005.

[5] F. Sebastiano, S. Drago, L. Breems, D. Leenaerts, K. Makinwa, and B. Nauta, "Impulse based scheme for crystal-less ULP radios," in *Proc. ISCAS*, May 2008, pp. 1508–1511.

[6] F. R. K. Soliman, S. Yuan, "An overview of design techniques for CMOS phase detectors," in *Proc. ISCAS*, May 2002, pp. 457–460.

[7] D. Wentzloff and A. Chandrakasan, "A 47pJ/pulse 3.1-to-5 GHz All-Digital UWB transmitter in 90nm CMOS," in *ISSCC Digest of Technical Papers.*, Feb. 2007, pp. 118–591.

[8] A. Abidi, "Phase Noise and Jitter in CMOS Ring Oscillators," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 8, pp. 1803–1816, Aug. 2006.