Post-Silicon Optimization of a Highly Programmable 64-MHz PLL Achieving 2.7-5.7 µW

Marco Gonzalez and David Bol
ICTEAM Institute, Université catholique de Louvain, Louvain-la-Neuve, Belgium
Email: {marco.gonzalez, david.bol}@uclouvain.be

***Abstract*—Hierarchical optimization methods used in the design of complex mixed-signal systems require accurate behavioral models to avoid the long simulation times of transistor-level SPICE simulations of the whole system. However, robust behavioral models that accurately model circuit non-idealities and their complex interactions must be very complex themselves and are hardly achievable. Post-silicon tuning, which is already widely used for the calibration of analog building blocks, is an interesting alternative to speed up the optimization of these complex systems. However, post-silicon tuning usually focuses on single-objective problems in blocks with a limited number of degrees of freedom. In this paper, we propose a post-silicon "hardware-in-the-loop" optimization method to solve multi-objective problems in mixed-signal systems with numerous degrees of freedom. We use this method to optimize the noise-power trade-off of a 64-MHz phase-locked loop (PLL) based on a back-bias-controlled ring oscillator. A genetic algorithm was run based on measurements of the 22-nm fully-depleted silicon-on-insulator prototype to find the Pareto-optimal configurations in terms of power and long-term jitter. The obtained Pareto front gives a range of power consumption between 2.7 and 5.7 µW, corresponding to an RMS long-term jitter between 88 and 45 ns. Whereas the simulation-based optimization would require more than a year using the genetic algorithm based on SPICE simulations, we conducted the post-silicon optimization in only 17 h.**

I. Introduction

The design of complex analog and mixed-signal systems has historically relied on hierarchical optimization methods to facilitate the task of designers. In these methods, the system is first divided into smaller functional blocks which can be designed at the transistor level, after which they are assembled to verify the system-level performance. Models of the behavior of the different sub-blocks are usually used along the design. The existing hierarchical methods can be separated into mainly three categories depending on the approach taken for the design of the sub-blocks and how these behavioral models are exploited. First, bottom-up methods design the sub-blocks at the transistor level before assembling them and verifying the system-level performance. It is at this last stage that the behavioral models replace the flat simulations of the complete system [1], [2], but they can lead to non-functional or sub-optimal systems because they do not consider system-level aspects in the design of the sub-blocks. Second, top-down methods derive the requirements of each sub-block from the system specifications to ensure its functionality [3]–[5]. The behavioral models are used to make this translation, which can sometimes lead to unachievable specifications for certain sub-blocks. Finally, hybrid methods, like the concurrent constraint-driven methodology [6], look to solve the limitations of the previous two approaches. In this method, the block-level parameters are optimized simultaneously with the system-level specifications for each sub-block. A numerical optimizer minimizes a cost function that not only includes the desired system-level performance metrics but also the difference between the modeled and the simulated behavior of the sub-blocks. The main disadvantage common to all these existing methodologies is that they rely on behavioral models of the different sub-blocks to decrease the simulation time compared to a transistor-level SPICE simulation. These models, however, can fail to reproduce precisely the behavior of complex blocks. The presence of circuit non-idealities and their complex interactions force the models to be complex. Additionally, some non-idealities cannot be known exactly at design time, such as the precise values of the parasitic elements or process variations.

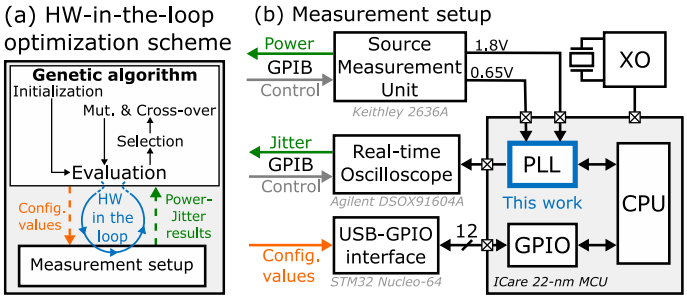

Fig. 1. (a) Hardware-in-the-loop optimization scheme based on a genetic algorithm. (b) Automatic measurement setup.

An alternative to avoid the long SPICE simulation times when designing the blocks would be to rely on measurements of the fabricated circuit, by doing a final post-silicon optimization. Post-silicon tuning is already a common technique used for analog circuits like current references [7]–[9] or relaxation oscillators [10], [11]. The main objective is to achieve the desired performance despite the fabrication process variations. The optimization of such circuits consists in trimming a component to calibrate the nominal current or frequency, or to minimize their dependence on the temperature. Both objectives are usually reached by modifying independently two degrees of freedom of the circuit [7], [10], [11]. Therefore, simple optimization algorithms, e.g., an exhaustive search, are sufficient for this task. Post-silicon tuning of more complex mixed-signal systems like

This work was supported by the Fonds de la Recherche Scientifique - FNRS under Grant n° CDR J.0014.20.

TABLE I
OPTIMIZATION TIME.

	Single perf. evaluation	Exhaustive search of the optimal solution	Genetic optimization
Simulation in the loop*	≈ 4 days	3.7×10^5 years	1.37 years
Hardware in the loop	20 s	21 years	17 h + fab. time

* SpectreX post-layout transient noise simulation on 8 cores at 2.4 GHz

phase-locked loops (PLLs) has also been proposed in [12], but the authors focus on improving the jitter of a PLL optimized at the design stage by compensating for a single non-ideality due to process variations. Generalizing these methods to multi-objective problems in such complex systems is anything but trivial. For instance, the jitter and the power consumption of a PLL must be co-optimized because these metrics are in direct conflict. Although the jitter of PLLs can be modeled relatively accurately and therefore optimized [13], [14], the trade-off between power and jitter is less straightforward, especially in the presence of circuit non-idealities [15], and it depends on numerous degrees of freedom. This makes an exhaustive search for finding the optimal points very time-consuming.

In this paper, we propose a post-silicon optimization method with the hardware in the loop and based on a genetic algorithm to solve multi-objective problems in complex analog and mixed-signal systems having multiple degrees of freedom. This method is used to optimize the noise-power trade-off of a 64-MHz PLL based on a back-bias-controlled ring oscillator, which was made highly programmable with five degrees of freedom. In addition, the method can be adapted to other performance metrics of PLLs, e.g., locking time, or to other use cases. The PLL was designed for the clock generation of a microcontroller unit (MCU) from a 32.768-kHz crystal oscillator (XO), commonly present in MCUs for generating the real-time clock. It was prototyped in an ultra-low-power (ULP) MCU codenamed ICare fabricated in 22-nm fully-depleted silicon-on-insulator (FD-SOI) technology. A Pareto front of the PLL noise-power trade-off was extracted from experimental measurements of the prototype, resulting in only 2.7 to 5.7 µW of power consumption for a long-term jitter between 88 and 45 ns.

This paper is organized as follows. Section II presents the proposed post-silicon optimization method. The design of the highly programmable PLL is discussed in Section III. Finally, measurement results are presented in Section IV, followed by a conclusion in Section V.

II. Proposed Post-Silicon Optimization Method

The different performance metrics of a charge-pump (CP) PLL, in particular its noise-power trade-off, depend on complex interactions between five degrees of freedom: the oscillator driving strength, the CP current, and the values of the three passive components of the low-pass filter (LPF). These complex interactions complicate the theoretical prediction of which combinations reach the optimal noise-power trade-offs of the PLL. Therefore, an optimization algorithm is preferable for finding these combinations. As shown in Table I, the time

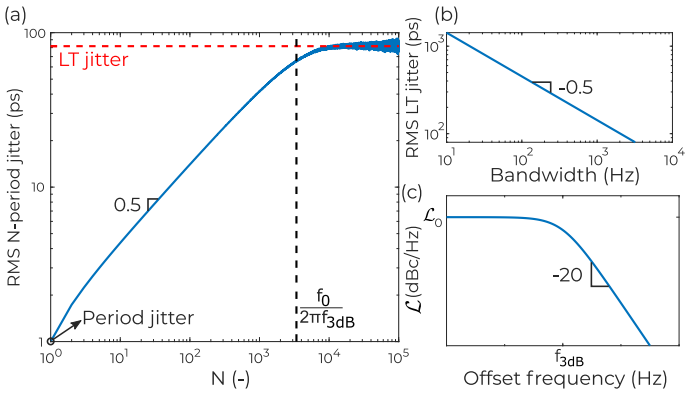

Fig. 2. Behavioral simulations: (a) N-period jitter profile for a 64-MHz PLL with a 3-kHz bandwidth and (b) effect of the 3-dB PLL bandwidth on the LT jitter. (c) Phase noise profile.

required for this optimization using SPICE transient noise simulations is prohibitive, particularly due to the large time intervals that need to be simulated because of the low-frequency reference. Using behavioral models for accelerating the simulations is an option [16] but because these complex interactions depend on the circuit non-idealities, the behavioral models need to be accurate enough and thus complex as well. Alternatively, we propose to perform the optimization post-silicon with the hardware in the loop by using the measurements from the fabricated prototype. For that, the PLL was made highly programmable by implementing a large range of values for the five degrees of freedom. Table I shows that this approach significantly reduces the optimization time.

Fig. 1(a) shows the general flow of the proposed method. The non-dominated sorting genetic algorithm (NGSA-II) [17] was selected as the multi-objective optimization algorithm for the noise-power trade-off of the PLL because it has been shown to perform well for analog sizing [18]. The algorithm is based on the evolution of a population for several generations. The population is made up of individuals having a different set of parameters and, at each generation, the objectives are measured and the fittest individuals are selected in an elitist approach. Genetic operators such as mutation and cross-over are used to create the next generation based on the fittest individuals.

A. Optimization Objective

The optimization objective for this work is the noise-power trade-off of the PLL, but the method can be easily adapted to include other performance metrics, e.g., the locking time. The noise in a PLL is quantified through two figures of merit, jitter in the time domain and phase noise in the frequency domain.

Jitter has various representations which need thus to be clarified. Period jitter is the random variation of the duration of a clock cycle. In free-running oscillators, it accumulates over time, but PLLs effectively stop the accumulation to keep the output clock synchronized with the reference clock. The accumulation of jitter is commonly characterized by the N-period jitter, defined as the deviation of N clock cycles from the duration of N nominal periods [13], [19]. Fig. 2(a) depicts the N-period jitter profile for a PLL. As the N-period jitter stabilizes

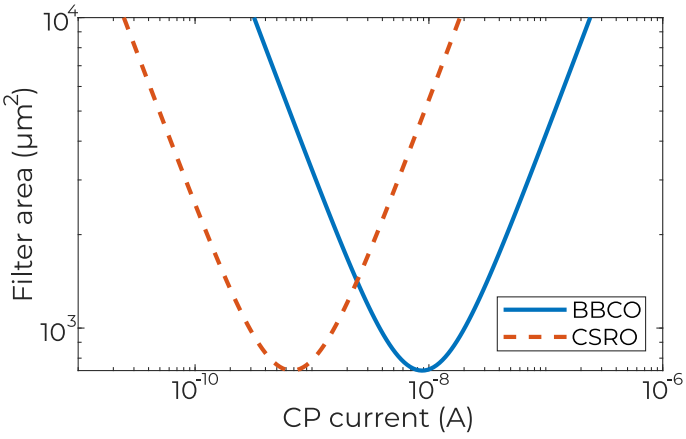

Fig. 3. Area of the conventional second-order passive low-pass filter in 22-nm technology for a PLL bandwidth of 3 kHz and a damping factor of $\sqrt{2}/2$. The considered capacitance and resistance densities are 9 fF/μm^2 and 125 kΩ/μm^2, respectively.

for large N values, we can define the long-term (LT) jitter as the N-period jitter when N tends to infinity.

The choice of the PLL bandwidth ($f_{3\text{dB}}$) directly impacts the jitter performance. A higher bandwidth will pass more reference-referred phase noise but a lower bandwidth will pass more phase noise, which will directly affect the period jitter of the PLL [19]. Moreover, a lower bandwidth allows more jitter accumulation over time for a fixed period jitter, resulting in a higher LT jitter [19]. In Fig. 2(b), we see that there is a clear relationship between the LT jitter and the square root of the bandwidth, evidenced by the slope of the line in log-log scale.

The frequency-domain counterpart of jitter is the phase noise ($\mathcal{L}$). The simplified phase noise profile of a PLL is shown in Fig. 2(c). It is linked to the LT jitter (σ_{LT}) through the following expression [19]:

$$\sigma_{\text{LT}} = \lim_{N\to\infty} \sqrt{\frac{\mathcal{L}_0 f_{3\text{dB}}}{2\pi f_0^2}\left(1 - e^{-2\pi f_{3\text{dB}} N/f_0}\right)} = \sqrt{\frac{\mathcal{L}_0 f_{3\text{dB}}}{2\pi f_0^2}}, \quad (1)$$

where f_0 is the operating frequency and $\mathcal{L}_0$ is the phase noise level at low frequencies. This expression shows that minimizing the LT jitter is equivalent to minimizing the phase noise. Therefore, the optimization of the PLL in this work will be done in terms of the LT jitter without loss of generality.

B. Implementation

Fig. 1(b) shows the automatic measurement setup implemented to run the optimization. To measure the objectives, i.e., power consumption and LT jitter, of each individual set of design parameters in a given generation of the genetic algorithm, the configuration values are communicated to the chip under test one at a time with an ad hoc protocol using the general-purpose I/Os (GPIOs). Each configuration value corresponds to a 25-bit word containing the values of the five degrees of freedom. An STM32 Nucleo-64 board is used as the interface between the serial communication with the computer running the optimization and the GPIOs of the chip. For the power measurements, the 2636A source measurement unit from Keithley provides both supply voltages needed in the PLL (i.e., 0.65 and 1.8 V) and measures the corresponding current.

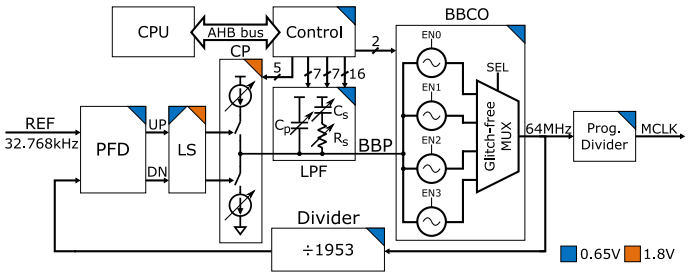

Fig. 4. Schematic of the proposed programmable PLL.

The DSOX91604A real-time oscillator from Agilent is used to measure the jitter, it outputs the N-period jitter as the standard deviation of 2×10^5 samples of the duration of 1×10^4 periods of the clock. All measurement results for a single individual (i.e., a single configuration of the PLL) are directly retrieved by the computer via GPIB, which triggers the next measurement by sending the new configuration to the chip. Once the measurements of all the individuals in the generation are done, the algorithm does the selection of the fittest individuals and creates the next generation using the genetic operators. The algorithm stops once a given number of generations have passed or a certain stopping criterion is achieved, e.g., the hypervolume metric [18].

III. Design of the Proposed Programmable PLL

A. Choice of Oscillator

The conventional architecture for ring oscillators is the current-starved architecture. However, back-bias-controlled oscillators (BBCOs), a novel architecture enabled by the forward and reverse back-bias capabilities of the FD-SOI technology, offer some advantages over the conventional architecture. They can achieve lower power consumption and slightly better performance in terms of the noise-power trade-off because of the shorter transistor stacks [20]. BBCOs also exhibit a more linear transfer function and thus a constant frequency gain over a wide PLL frequency range [20]. In the context of PLL design, this translates into phase noise reduction, and thus jitter, at the PLL output [21].

The silicon area of the low-pass filter (LPF) is an issue in charge-pump PLLs with low reference frequencies because of the large capacitors needed to obtain a low bandwidth, which is required for stability [22]. It thus needs to be minimized to avoid the need for off-chip components. As shown in Fig. 3, when a fixed bandwidth and damping factor of the PLL are considered, the minimum area occupied by the LPF is achieved at a higher charge pump (CP) current for the BBCO than for the current-starved ring oscillator (CSRO). This is because BBCOs have a frequency gain that is typically 10-20× lower than that of CSROs [20]. This increase in CP current remains acceptable in terms of power consumption because it does not actuate continuously and its power remains low compared to the BBCO, which is the greatest contributor to the PLL power. The advantage of this current increase is that it decreases the phase noise due to the CP because its noise is scaled by the squared nominal value of the current [13].

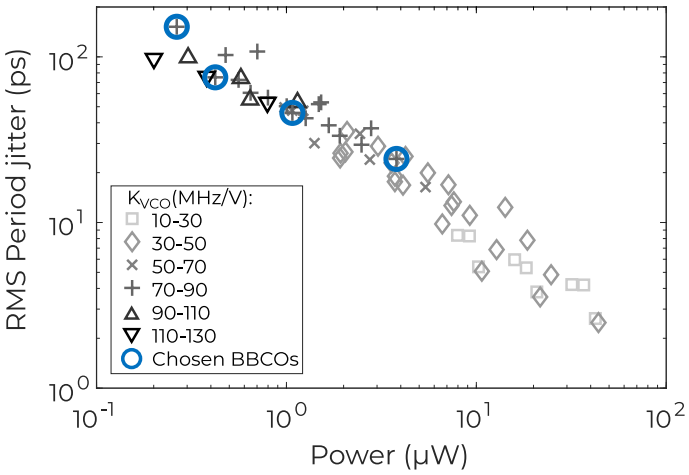

Fig. 5. Noise-power trade-off for different BBCOs grouped according to their frequency gain. A four-point Pareto front highlights the selected oscillators.

Given the low frequency of operation, we use ultra-high-V_t (UHVT) transistors in the design of the BBCO to reduce its length and minimize its area. This transistor flavor has reverse back-bias (RBB) capabilities, i.e., in terms of absolute values, their threshold voltage can be increased when increasing their back-bias voltages. Therefore, these voltages can be used to control the oscillation frequency. However, negative voltages are needed to control the body of NMOS transistors in the case of RBB. Generating negative voltages for driving the BBCO can be done efficiently with switched-capacitor CPs for a few µW [23]. However, this power overhead in the µW range is not negligible in this design. Therefore, to avoid the area and power overhead of generating a negative voltage, only the back-bias voltage of the PMOS transistors (BBP) was kept as a tuning knob for the oscillator frequency. Simulations show that this degrades the duty cycle of the output clock signal by less than 2 %.

B. PLL Architecture

Fig. 4 shows the schematic of the proposed PLL. It is part of a ULP MCU codenamed ICare where the logic and SRAM memories are supplied at 0.65 V and the I/O voltage is 1.8 V. Three main modifications were introduced to the conventional architecture for controlling the BBCO. Firstly, although transistors in this technology can withstand up to 3 V on the back gate, this range was reduced to 1.8 V to comply with the I/O voltage of the MCU. A level shifter (LS) was added at the interface of the charge pump and the phase-frequency detector (PFD), which works at 0.65 V as the rest of the blocks. Finally, the third modification was done to the LPF to limit the impact of supply noise on the oscillator. An AC coupling between the BBP node and the 0.65-V BBCO supply was added by modifying the filter to have the capacitors connected to the supply instead of the ground.

C. BBCO Programmability

The inherent noise-power trade-off of the BBCO can be exploited to make the PLL programmable. At a fixed oscillation

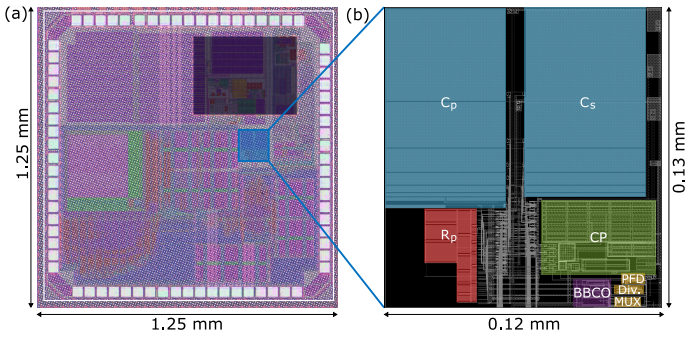

Fig. 6. (a) Chip microphotograph with overlaid layout and (b) layout of the proposed PLL.

TABLE II
RANGE OF THE DEGREES OF FREEDOM.

Parameter	Range
BBCO period jitter	24 to 152 ps
Charge pump current	5 to 160 µA
Series capacitance (C_s)	0.25 to 32 pF
Series resistance (R_s)	1 to 64 MΩ
Parallel capacitance (C_p)	0 to 32 pF

frequency, increasing the number of inverting stages in parallel in a ring oscillator is a good way to decrease the phase noise and the jitter, at the cost of an increased power consumption [20]. Nevertheless, the addition of switches at the input and/or output of the inverting stages has the risk of degrading the phase noise. Instead, we chose to design a bank of ring oscillators with different noise-power trade-offs. As can be seen in Fig. 4, four BBCOs were implemented in parallel and multiplexed through a glitch-free multiplexer. Each of them is disabled when it is not used to avoid power overheads. As UHVT transistors are used, the leakage overhead of implementing multiple BBCOs is negligible.

For selecting these four BBCOs, oscillators composed of different standard cells were simulated to select those with the best performance in terms of power, period jitter, and frequency gain. Inverter gates as well as higher fan-in inverting gates, i.e., NAND and NOR gates with short-circuited inputs, were considered. The results are shown in Fig. 5. The points are grouped according to their frequency gain (K_{VCO}). For the PLL design, the selected oscillators must have similar frequency gains to avoid a large range for the LPF parameters needed to stabilize the system. Between the groups of oscillators with the highest number of points, the group whose gain is between 70 and 90 MHz/V was selected because a higher frequency gain is preferable to minimize the CP current and the LPF area. Inside this group, four optimal evenly-distributed points were selected in our implementation. They are highlighted in Fig. 5.

D. Bandwidth Programmability

Another way to add programmability to the PLL is through the bandwidth, which can be modified by changing the CP current or the values of the LPF passive components. Changing the bandwidth contributes to the noise-power trade-off in the following way. A smaller bandwidth can be achieved by decreasing the CP current, thus lowering the peak power con-

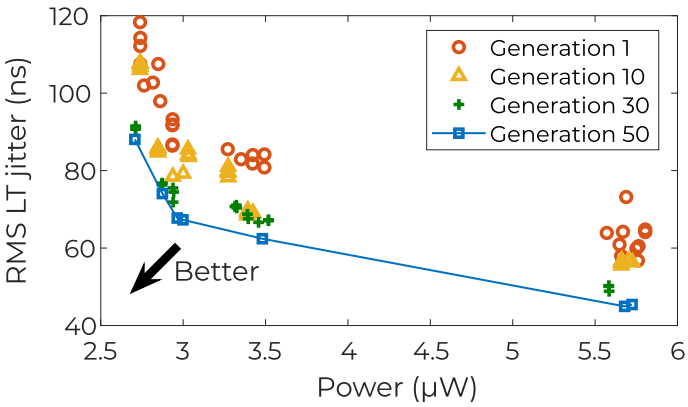

Fig. 7. Pareto front of the noise-power trade-off across 50 generations of the NGSA-II algorithm.

sumption. As depicted in Fig. 2(b), a smaller bandwidth implies a higher LT jitter because it accumulates for a longer time. On the contrary, a higher bandwidth makes the accumulation stop sooner but needs a higher CP current.

Table II summarizes the degrees of freedom and their corresponding range. A 5-bit current digital-to-analog converter controls the CP current. Two parallel 7-bit binary banks of capacitors were implemented for C_s and C_p. Finally, a series bank of resistors with logarithmically-spaced values was implemented for R_s. The value was controlled through 16 bits with a one-hot code.

IV. Measurement Results

The proposed circuit was prototyped in Global Foundries 22-nm FD-SOI technology. As shown in Fig. 6, it occupies an area of 0.0156 mm^2 on the 1.25×1.25 mm ICare Cortex-M4 MCU chip.

Fig. 7 shows the process of convergence across 50 generations from an initially random population. The only performance constraint in the algorithm concerns the bandwidth of the PLL. A PLL with a bandwidth higher than 10 % of the reference frequency is not able to filter correctly the signal at the output of the PFD, thus resulting in non-optimal results. However, the values of the filter components are particularly subject to process variations, making the analytical approximation of the bandwidth not precise. Therefore, the constraint was finally relaxed by a factor of two. In the obtained Pareto front, the point with the best power consumes 2.7 µW for an LT jitter of 88 ns, whereas the point with the best LT jitter is at 45 ns and consumes 5.7 µW.

The values of three of the five degrees of freedom for the optimal points in the Pareto front are shown in Fig. 8(a)-(c). The BBCO index corresponds to the digital code used to select one of the four BBCOs highlighted in Fig. 5, ordered in ascending order in terms of power consumption. The capacitances C_s and C_p are not shown because their selected values are shared by all the optimal points. C_s is fixed at 31 pF, whereas C_p relies only on the parasitic capacitances of the BBP node, estimated at 41 fF from post-layout parasitic extraction. Fig. 8(d) shows the PLL bandwidth resulting from these values, computed analytically. A linear dependence between power consumption and

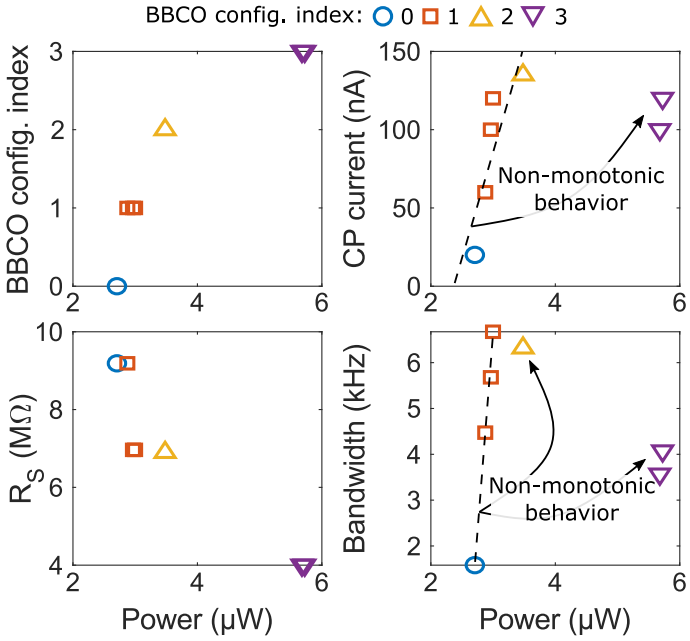

Fig. 8. Values of the degrees of freedom for the optimal points in the Pareto front: (a) digital code to select the BBCO ordered in ascending order in terms of power consumption, (b) charge pump current, and (c) series resistance of the low-pass filter. (d) Analytically approximated PLL bandwidth.

PLL bandwidth can be appreciated for the lower-power points with a higher LT jitter. For fixed values of the capacitances, higher bandwidths are obtained by increasing the CP current. This results in a reduced jitter because the BBCO contributes less to the output noise, at the price of a higher power. Nevertheless, once the maximum bandwidth is reached, the relation becomes more complex and non-monotonic for the lower-jitter points. This shows the difficulty of predicting analytically the optimal design points in the noise-power trade-off of a PLL, motivating the need for post-silicon optimization.

It is also interesting to highlight that the BBCO dominates the noise-power trade-off of the PLL. The points that are near to others in the Pareto front share the same BBCO and only differ in the other filter parameters. To find other optimal points, a larger choice of oscillators must be included in the design. This is particularly true for points in the lower-jitter extreme of the Pareto front, as opposed to the lower-power extreme where the power of the rest of the blocks imposes the lower limit achievable in terms of power consumption.

Table III presents a comparison to recent state-of-the-art MHz-range PLLs. After optimization, the proposed PLL exhibits a normalized power 5× lower than the design in [24], which generates the same output frequency. Furthermore, the figure of merit (FoM) that allows comparing the power-jitter trade-off is 1.2 to 5.3 dB better, even though we use a reference frequency 15× lower, which tends to let the jitter accumulate more. Lower reference frequencies limit the PLL bandwidth and thus the filtering of the oscillator phase noise. The designs in [25], [26], which also use a 32-kHz reference but generate an output frequency 2.9 to 4.8× higher, exhibit a better phase noise. However, they rely on more complex active low-pass filter architectures which have a power overhead resulting in a

TABLE III
COMPARISON TO STATE-OF-THE-ART MHZ-RANGE PHASE-LOCKED LOOPS.

	This work	JSSC'19 [24]	ESSCIRC'16 [25]	TCAS-II'12 [26]
Technology	22 nm FD-SOI	65 nm Bulk	28 nm FD-SOI	45 nm Bulk
Supply voltage [V]	0.65/1.8	0.8	0.5/1.5	-
Frequency [MHz]	64	64	307.2	184
Ref. frequency [kHz]	32.768	500	32.768	32.768
N. power [nW/MHz]	42 to 89	466	1953	5978
Settling time [µs]	400*	-	2330	-
$\mathcal{L}$(10 kHz) [dBc/Hz]	−21 to −29	−57	−62	−48
$\mathcal{L}$(1 MHz) [dBc/Hz]	−83	−77	−110.6	−105.2
RMS period jitter [ps]	99.6 to 120.8	55.3	-	-
FoM [dB]†	−221.6 to −225.7	−220.4	-	-
Area [mm^2]	0.0156	0.016	0.0175	0.086

* After change of configuration
† FoM$= 20\log\left(\frac{\text{Period jitter}}{1\,\text{s}}\right) + 10\log\left(\frac{\text{Power}}{1\,\text{mW}}\right)$

normalized power 29 to 142× higher.

V. CONCLUSIONS

In this work, we design a highly programmable 64-MHz PLL for digital clock generation based on a back-bias-controlled ring oscillator and optimize it with a post-silicon "hardware-in-the-loop" optimization scheme based on a genetic algorithm. The optimization scheme was run on an automatic setup for the measurement of the power consumption and the jitter of the PLL. The resulting Pareto-optimal configurations yield a power consumption range from 2.7 to 5.7 µW, corresponding to an RMS LT jitter from 88 to 45 ns. The evolution of the five degrees of freedom across the Pareto front shows a non-monotonic behavior for the low-jitter points, highlighting the difficulty of predicting them analytically and the advantages of post-silicon optimization. Furthermore, the optimization time was decreased with the proposed scheme to only 17 h, in contrast to the simulation-based alternative that would require more than a year.

ACKNOWLEDGMENTS

The authors would like to thank Pierre Gérard from the UCLouvain WELCOME platform for the measurement setup.

REFERENCES

[1] G. Gielen, T. McConaghy, and T. Eeckelaert, "Performance space modeling for hierarchical synthesis of analog integrated circuits," in *Des. Automat. Conf.*, 2005, pp. 881–886.

[2] R. A. Rutenbar, G. G. E. Gielen, and J. Roychowdhury, "Hierarchical Modeling, Optimization, and Synthesis for System-Level Analog and RF Designs," *Proc. IEEE*, vol. 95, no. 3, pp. 640–669, Mar. 2007.

[3] B. G. Arsintescu, E. Charbon, E. Malavasi, U. Choudhury, and W. H. Kao, "General AC constraint transformation for analog ICs," in *Des. Automat. Conf.*, 1998, pp. 38–43.

[4] Y. Fellah, N. Abouchi, T. Tixier, A. Aubert, and L. Labrak, "Top-down, constraint driven design : Application to sigma-delta modulator," in *2004 IEEE Int. Conf. on Ind. Technol.* IEEE, 2004, pp. 132–136.

[5] S. Dam and P. Mandal, "Modeling and design of CMOS analog circuits through hierarchical abstraction," *Integration*, vol. 46, no. 4, pp. 449–462, Sep. 2013.

[6] R. Phelps, M. J. Krasnicki, R. A. Rutenbar, L. R. Carley, and J. R. Hellums, "A case study of synthesis for industrial-scale analog IP: Redesign of the equalizer/filter frontend for an ADSL CODEC," in *Des. Automat. Conf.*, 2000, pp. 1–6.

[7] F. Crupi, R. D. Rose, M. Paliy, M. Lanuzza, M. Perna, and G. Iannaccone, "A portable class of 3-transistor current references with low-power sub-0.5 V operation," *Int. J. of Circuit Theory and Appl.*, vol. 46, no. 4, pp. 779–795, 2017.

[8] Q. Huang, C. Zhan, L. Wang, Z. Li, and Q. Pan, "A -40 °C to 120 °C, 169 ppm/°C Nano-Ampere CMOS Current Reference," *IEEE Trans. Circuits Syst. II*, vol. 67, no. 9, pp. 1494–1498, Sep. 2020.

[9] M. Lefebvre and D. Bol, "A Family of Current References Based on 2T Voltage References: Demonstration in 0.18-μm With 0.1-nA PTAT and 1.1-μA CWT 38-ppm/°C Designs," *IEEE Trans. Circuits Syst. I*, vol. 69, no. 8, pp. 3237–3250, Aug. 2022.

[10] H. Jiang, P.-H. P. Wang, P. P. Mercier, and D. A. Hall, "A 0.4-V 0.93-nW/kHz Relaxation Oscillator Exploiting Comparator Temperature-Dependent Delay to Achieve 94-ppm/°C Stability," *IEEE J. of Solid-State Circuits*, vol. 53, no. 10, pp. 3004–3011, 2018.

[11] J. Mikulic, G. Schatzberger, and A. Baric, "Post-Manufacturing Process and Temperature Calibration of a 2-MHz On-Chip Relaxation Oscillator," *IEEE Trans. on Circuits and Syst. I: Regular Papers*, vol. 68, no. 10, pp. 4076–4089, 2021.

[12] G. Yu and P. Li, "Hierarchical Analog/Mixed-Signal Circuit Optimization Under Process Variations and Tuning," *IEEE Trans. on Computer-Aided Des. of Integr. Circuits and Syst.*, vol. 30, no. 2, pp. 313–317, 2011.

[13] M. Perrott, M. Trott, and C. Sodini, "A modeling approach for Σ-Δ fractional-N frequency synthesizers allowing straightforward noise analysis," *IEEE J. Solid-State Circuits*, vol. 37, no. 8, pp. 1028–1038, Aug. 2002.

[14] B. Razavi, "Jitter-Power Trade-Offs in PLLs," *IEEE Trans. Circuits Syst. I*, vol. 68, no. 4, pp. 1381–1387, Apr. 2021.

[15] ——, "Phase-locked loops," in *Design of Analog CMOS Integrated Circuits*. McGraw-Hill, 2001.

[16] J. Zou, D. Mueller, H. Graeb, and U. Schlichtmann, "A CPPLL hierarchical optimization methodology considering jitter, power and locking time," in *Des. Automat. Conf.*, 2006, pp. 19–24.

[17] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimization: NSGA-II," in *Int. Conf. on Parallel Problem Solving from Nature*, 2000, pp. 849–858.

[18] R. Dekimpe and D. Bol, "A Configurable ULP Instrumentation Amplifier With Pareto-Optimal Power-Noise Trade-Off Achieving 1.93 NEF in 65nm CMOS," *IEEE Trans. on Circuits and Syst. II: Express Briefs*, vol. 68, no. 7, pp. 2272–2276, 2021.

[19] N. Da Dalt and A. Sheikholeslami, *Understanding Jitter and Phase Noise: A Circuits and Systems Perspective*, 1st ed. Cambridge University Press, 2018.

[20] M. Schramme, L. Van Brandt, D. Flandre, and D. Bol, "Comprehensive Analytical Comparison of Ring Oscillators in FDSOI Technology: Current Starving Versus Back-Bias Control," *IEEE Trans. Circuits Syst. I*, 2022.

[21] E. Ayranci, K. Christensen, and P. Andreani, "Enhancement of VCO Linearity and Phase Noise by Implementing Frequency Locked Loop," in *The Int. Conf. on "Computer as a Tool"*. IEEE, 2007, pp. 2593–2599.

[22] A. Lahiri, N. Gupta, A. Kumar, and P. Dhadda, "A 600µA 32 kHz input 960 MHz output CP-PLL with 530ps integrated jitter in 28nm FD-SOI process," in *40th Eur. Solid-State Circuits Conf. (ESSCIRC)*. IEEE, Sep. 2014, pp. 87–90.

[23] R. Dekimpe, M. Schramme, M. Lefebvre, A. Kneip, R. Saeidi, M. Xhonneux, L. Moreau, M. Gonzalez, T. Pirson, and D. Bol, "SleepRider: A 5.5μW/MHz Cortex-M4 MCU in 28nm FD-SOI with ULP SRAM, Biomedical AFE and Fully-Integrated Power, Clock and Back-Bias Management," in *2021 Symp. on VLSI Circuits*. IEEE, Jun. 2021.

[24] J. Zhu, W.-S. Choi, and P. K. Hanumolu, "A 0.016 mm^2 0.26-µW/MHz 60–240-MHz Digital PLL With Delay-Modulating Clock Buffer in 65 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 54, no. 8, pp. 2186–2194, Aug. 2019.

[25] A. Lahiri and N. Gupta, "A 0.0175mm ^{2} 600µW 32kHz input 307MHz output PLL with 190psrms jitter in 28nm FD-SOI," in *42nd Eur. Solid-State Circuits Conf.* IEEE, Sep. 2016, pp. 339–342.

[26] Xiao Pu, A. Kumar, and K. Nagaraj, "Area-Efficient Low-Noise Low-Spur Architecture for an Analog PLL Working From a Low Frequency Reference," *IEEE Trans. Circuits Syst. II*, vol. 59, no. 6, pp. 331–335, Jun. 2012.