A Double-Edge Triggered Asynchronous Gray Counter for Use as a Coarse Counter in VCO ADCs

Johan Raman, *Member, IEEE*, Tobias Cromheecke, Simon Ooghe, Brendan Saux, and Pieter Rombouts, *Senior Member, IEEE*

***Abstract*—This brief presents an asynchronous double-edge triggered Gray counter that can advantageously be used as coarse counter in a coarse-fine VCO ADC. We show that, in spite of using only a single counter and a single sampling register, it is possible to resolve the ambiguity in the coarse counter value caused by metastability and delay. We present simple decoding logic that performs both Gray-to-binary conversion and ambiguity resolution. A comparative case study is made for an 11 bit coarse-fine architecture based on the new Gray counter and a prior-art double-sampling binary counter. A reduction of 21% in area has been obtained, while at the same time post-layout simulations show a power saving of 17.5%.**

***Index Terms*—Analog-to-digital conversion (ADC), ring oscillators (RO), voltage-controlled oscillators (VCOs), frequency conversion, VCO ADC, coarse-fine architecture, Gray counter, double sampling.**

I. Introduction

ANALOG-TO-DIGITAL converters (ADCs) are ubiquitous in electronic systems. Within the spectrum of known architectures, ADCs based on voltage-controlled oscillators (VCOs) are relatively young [1], [2], [3]. Strong points of VCO ADCs are their digital-oriented nature, small size, inherent anti-aliasing and noise shaping. Also in terms of power efficiency, important progress is being made [4], [5]. Especially the so-called coarse-fine architecture [6], illustrated in Fig. 1, is paving the way to more power-efficient VCO ADCs [7], [8], [9]. A key advantage of the coarse-fine architecture is that the sample rate f_s can be much lower than the VCO frequency f_{VCO}, which translates into a reduced dynamic power consumption for the readout circuits working at this lower sample rate. However, the coarse counter itself still operates at the relatively high f_{VCO} frequency, and so its dynamic power consumption is of importance.

Coarse-fine VCO architectures also have important issues [10]. Two closely related problems are counter state metastability and counter delay. The former problem occurs during incrementing of the coarse counter, where the state during transition is ill-defined, possibly leading to large errors when sampling the counter state at these instants. The latter problem causes a time skew between the coarse and fine states, potentially causing an error in the capture of the coarse counter state with one count. Both of these problems relate to a broader asynchronicity problem caused by the VCO-driven counter and the sampling circuits being in different clock domains.

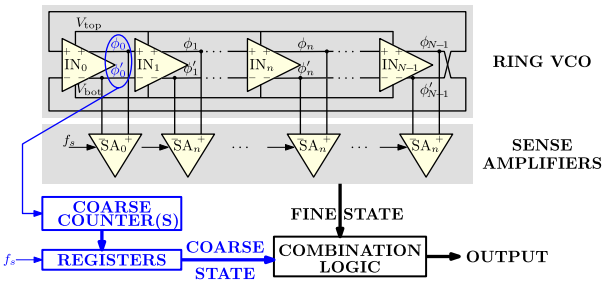


Fig. 1. Coarse-fine architecture, consisting of a ring VCO, an array of sense amplifiers to sample the VCO state, a coarse counter that counts the number of cycles, registers to sample the counter state and combination logic.

A known solution to the above problems is the use of double sampling. There are a few variants for this. In a first variant, two coarse counters are used, one that updates with the reference phase, and one that updates with an opposite phase [9], [11]. The sampled VCO phase (i.e., the fine state) is used to determine which counter state was stable at the moment of sampling. In a second variant, there is only one counter, but its state is delayed by half a VCO cycle [12], [13]. Both the original and the delayed value are sampled, and an arbitration similar to what is needed for the first variant is used. A third variant is based on using a single counter, but three sampling registers [14]. An obvious disadvantage of double sampling is that area and/or power is at least doubled [15]. In present brief we show how this doubling can be avoided by presenting a new solution to the metastability and delay problem based on a single counter and a single sampling register.

There are some other architectural approaches that deal with the asynchronicity problem. In [7], the coarse and fine states are synchronized using a data scrambling technique, while in [8] the VCO phases drive a maximum length sequence generator. Both [7] and [8] rely on additional flip-flops (FFs) that are clocked at the speed of the VCO.

For completeness, we mention also work that relates to capturing the VCO state itself. In [16] potential errors when sampling the fine state are discussed, which is an aspect

Manuscript received 16 January 2024; revised 15 March 2024; accepted 2 April 2024. Date of publication 12 April 2024; date of current version 4 September 2024. This work was supported in part by VLAIO, and in part by Melexis. This brief was recommended by Associate Editor S. Pietri. *(Corresponding author: Johan Raman.)*

The authors are with the Department of Electronics and Information Systems, Ghent University, 9052 Gent, Belgium (e-mail: Johan.Raman@UGent.be).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSII.2024.3388104.

Digital Object Identifier 10.1109/TCSII.2024.3388104

that is orthogonal to our work. In [17], a Gray-encoded ring oscillator is presented in which the ring VCO signals directly drive Boolean logic, which on its turn operates toggle-mode FFs. While this approach has some similarity to our work, an important difference is that [17] does not represent a coarse-fine architecture. Therefore, the counter in [17] operates at a much higher rate and thus consumes more power.

The power-efficiency of a counter can be improved on two fronts: (i) minimizing the required Boolean logic, and (ii) avoiding unnecessary FF activations. A binary ripple counter [9], [13] is optimal with respect to the former aspect, because no Boolean logic is required at all. However, a binary counter has on the average two bit-transitions per count step. Theoretically, only a single transition per count step is needed. The core idea that led to present brief is the recognition that a Gray counter can be about 2 times more efficient on this aspect. However, this gain will partly be offset by the logic functions that are needed to activate the right FF.

The advantages of using a Gray counter instead of a binary counter have been recognized before [7], [15], [18], such as preventing erroneous intermediate states. However, to the authors' knowledge, prior art Gray counters used as coarse counters are synchronous [7], [15], [18] and thus require a substantial amount of Boolean functions or, in the case of [18], nearly double the number of FFs. An important contribution of our work here is that we present an asynchronous version of the Gray counter. The counter operates in such a way that only the single FF that needs to toggle is activated. The counter is also double-edge triggered w.r.t. the reference VCO phase that it receives. We will show later on that this can be exploited to resolve the one-off ambiguity that remains due to asynchronicity of the sampling clock.

The rest of this brief is organized as follows. The asynchronous Gray counter structure is presented in Section II. Resolving the one-off ambiguity is discussed in Section III (both method and circuit embodiment). Section IV highlights simulation results that validate the approach. Section V summarizes our conclusions.

II. Asynchronous Gray Counter

The proposed asynchronous Gray counter is displayed in Fig. 2. For reasons of clarity we limit our discussion here to a 6-bit counter, but the structure can be easily generalized to other bit lengths. The core of the Gray counter consists of an array of toggle-mode FFs, see Fig. 2a. Each of these FFs have a separate clock signal, denoted as clk_i with $i = 0, 1, \ldots, 5$. Because the FFs do not share a common clock, it is clear that the counter operates in an asynchronous way. The toggle-mode FFs are activated at the falling edge of their clock input. To minimize dynamic current consumption, only one of the clk_i signals exhibits a falling edge, which implements the feature discussed above that only a single FF is activated for each count step.

The circuit that generates the different clock signals clk_i is shown in Fig. 2b. The main input signal that drives this circuit is the reference phase ϕ_{ref} coming from the VCO. In the left branch of the circuit, the input ϕ_{ref} is simply inverted to obtain clk_0. Therefore, during a rising edge of ϕ_{ref}, clk_0 will go low and the corresponding falling edge will activate the toggling

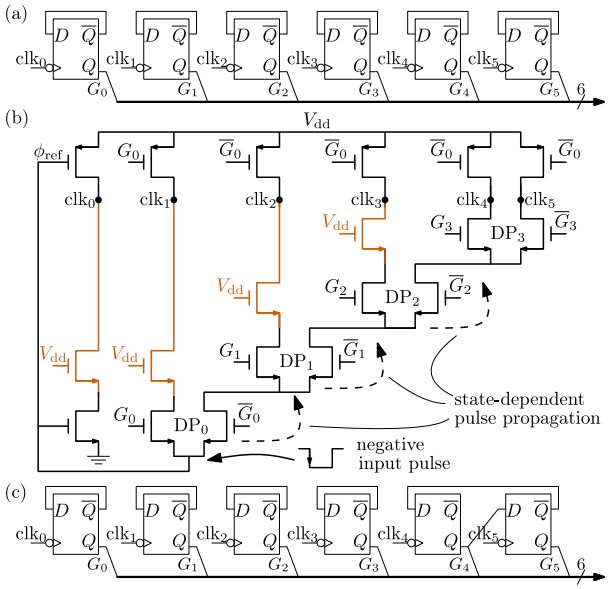


Fig. 2. (a) proposed 6-bit Gray counter based on toggle-mode flip-flops, (b) pulse distribution circuit, (c) Gray counter with cross-coupled end stage.

of G_0. As a result, the value of the counter is increased by one unit at the rising edge of ϕ_{ref}. At the next falling edge of ϕ_{ref}, the counter value will again be increased with one unit, i.e., the proposed counter is double-edge triggered. At this point, one of the G_i with $i > 0$ needs to toggle. The mechanism used here is that the negative pulse that starts with the falling edge of ϕ_{ref} is propagated in a state-dependent way to the right clock signal clk_i (with $i > 0$). The point where this negative pulse enters is at the common-source point of the differential pair DP_0 (note it is connected to ϕ_{ref}). Suppose now that G_0 is high. Then the negative pulse will be propagated to clk_1, which is pulled down, leading then to G_1 toggling its value. If on the other hand, G_0 is low, the negative pulse will be propagated to the common source of differential pair DP_1. Depending now on the value of G_1, the negative pulse will either result in a downwards activation of clk_2, or it will be steered to DP_2. This pattern is then repeated for the next bits, hence G_2 steers the pulse to clk_3, and G_3 to clk_4. For the last possible activation site, in our case clk_5, we do not need to involve the state any more. When the pulse has traveled this far, it can be sent directly to the last FF, which is the only remaining one that can be toggled. So, the values of G_4 and G_5 are not needed for steering the activation pulse to the right FF. Finally, each of the branches that defines clk_i for $i > 0$ has a PMOS transistor that serves as a reset. In this, clk_1 is reset when G_0 is zero, i.e., when there is no pulse expected for this branch. Likewise, all other branches are reset when G_0 is one, i.e., when the next pulse will be directed to clk_1.

In order to prevent corruption of the clk signals, direct feedback of a toggling FF output to its own clk input should be avoided. This is guaranteed in the proposed pulse propagation network by making the clk_i signal dependent only on the state G_j with $j < i$. Consider for instance the situation $G_0 = 0$ and $G_1 = 1$. An incoming negative input pulse will then be directed to clk_2. As a result, the state G_2 will start to change, but this only affects the switching occurring in DP_2. Because

G_1 remains the same, DP_1 continues to conduct on the left side, while the right side (going to the affected differential pair) remains open.

The asynchronicity of the counter implies that the count delay becomes to some extent state dependent. This is clear from the pulse propagation circuit, since a pulse that needs to travel further also experiences more delay. For this reason, extra NMOS transistors (colored in Fig. 2b) can be inserted in the shorter propagation paths for the purpose of equalizing the delay of the various paths.

Setting the initial condition of the counter can be challenging. The counter can be forced in a zero state at startup by using resettable FFs. When after the reset there is first a rising edge of ϕ_{ref}, G_0 is toggled from zero to one, and the counter is started in the intended way. However, when a falling edge would come first, this edge would propagate to the last FF (because of the zero state of the counter, pulses are always sent to the right side). In this case, the last FF G_5 would be flipped from zero to one, which is not the intended startup behavior. It can be understood that the counter then actually starts to count backwards. In order to enforce that the counter operates in the upcounting mode, the wiring of the two highest bits is modified. This is shown in Fig. 2c. The last two FFs are no longer wired as individual toggle FFs, but rather operate as a two-bit Johnson counter. It can be shown that this rewiring does not affect the Gray code being generated. However, if a pulse would travel to clk_5 directly after a reset of the counter, G_5 will take over the value of G_4, which is zero. Hence, the counter remains in the zero state, and is ready to start counting upwards at the next rising edge of ϕ_{ref}. In practice, an explicit reset is not required. While the counter might initially count in the wrong direction, it is guaranteed to auto-correct at the latest when passing through the zero-state.

III. Resolving One-Off Ambiguity

We now return to the important problem of counter asynchronicity. With the use of a Gray counter, the code changes only in one position and hence the problem of erroneous intermediate states discussed in the introduction is avoided. Still, there is an important delay τ between the edges of the VCO reference phase ϕ_{ref} and the subsequent response of the counter. In this section we explain how we can deal with the one-off ambiguity that may exist when sampling the counter state due to the delay. In order to explain the core concept of the proposed solution, we use an idealized timing diagram of the counter as detailed in Fig. 3. In this, the top line represents the input VCO phase ϕ_{ref}. Subsequent lines are the counter FF states going from G_0 up to G_5. The last line is the theoretical modulo-2 sum of the counter bits, i.e., the parity of the counter value. Note that this parity also corresponds with the least significant bit when converting the Gray code into a binary representation. The idealization is that we assume all propagation delays to be identical, i.e., the delays for the different bit transitions are perfectly equalized. We now consider an arbitrary sampling instant. At the sample instant, both the counter and the VCO fine state are determined. Because ϕ_{ref} is just an ordinary VCO phase, a sampled value of it will be known as part of the fine state.

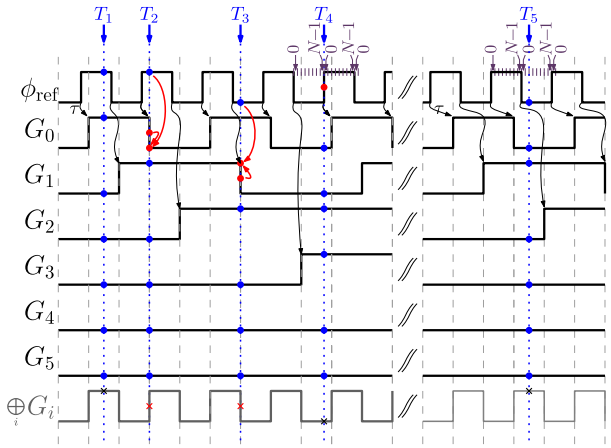


Fig. 3. Simplified timing diagram to explain the method for resolving the one-off ambiguity. The last line is the instantaneous parity of the counter. In the left part, the delay τ is rather small, while on the right part, τ is getting close to half a VCO period.

The rationale of the ambiguity resolution method is very easy to grasp when comparing the reference phase ϕ_{ref} to the instantaneous parity of the Gray code. They are, up to the delay τ of the counter, identical. Since an ideal counter would have τ equal to zero, we can see that the sample being taken of ϕ_{ref} corresponds with the expected parity of the Gray counter. Based on this we can easily correct the counter value: if the parity of the captured counter value matches with the ϕ_{ref} sample value the counter state is right, while if it does not, the counter state needs to be incremented with one unit. Note that the parity of the captured counter value can be determined with simple Boolean logic that operates at the relatively low sample rate f_s. Putting now things together, the digital output can be easily obtained from the fine state, that ranges from 0 to $N-1$ within each half period of ϕ_{ref} (N being the number of delay elements in the ring), and the corrected coarse count according to:

$$D_{\text{out}} = (\text{fine state value}) + (\text{corrected coarse count}) \times N \quad (1)$$

Let us visualize how the parity check allows to correct for ambiguity at different sampling instants (see Fig. 3). At the sampling instant T_1 the ϕ_{ref} sample value is one, indicating that the coarse counter state is expected to have a parity equal to one. The actual parity of the sampled counter value can be obtained from the Boolean post-processing circuit. In this case, the high value that is indicated on the lower trace of the figure will be obtained. Since in this case the calculated parity matches with the expected parity, the counter value is recognized as being correct. Turning now to sample instant T_2, we see that this time G_0 is being sampled at a transition. In this case, the sample value of G_0 could turn out to be the old value (one) or the new value (zero). However, the ϕ_{ref} sample value is one, and so it is known that the expected parity of the counter value is one as well. In case the new value (zero) of G_0 had been captured, the parity will match, and so the counter value is validated as being correct. If instead the old value (one) of G_0 had been captured, the parity will not match, and so the counter value will be incremented, leading again to the correct result. A similar situation arises at sample instant

T_3, but this time it is G_1 that is causing the ambiguity. Note that the correction algorithm does not need to know which FF state is ambiguous. The important element is that, thanks to the Gray encoding, only one bit can be ambiguous, and that the expected parity allows to resolve the correct counter value. A last sample instant of interest is T_4. Here, the sampling of the reference phase ϕ_{ref} is ambiguous. To explain that the correction also works in this situation, it is important to take the fine state value into account. When the ϕ_{ref} sample is zero, the fine state value is $N-1$ (see the fine count scale added in the figure). Since the course counter parity is zero as well, it is concluded that the coarse counter value is correct. With this equation (1) becomes $D_{\text{out},1} = (N-1) + (\text{coarse count}) \times N$. If however the ϕ_{ref} sample is one, the expected parity does no longer match with the parity as determined from the counter state. Hence, the counter will need to be corrected with one unit. The associated fine state value is in this situation however 0. Therefore, the digital output value becomes $D_{\text{out},2} = 0 + (\text{coarse count} + 1) \times N$. The difference between $D_{\text{out},2}$ and $D_{\text{out},1}$ is one, which is expected because it reflects the fact whether the ϕ_{ref} transition occurring at the sample instant has been included or not. So, also for sampling instant T_4 the correction method works.

However, the range of allowed delays τ is limited by certain factors. It can for instance be noted that the above method for ambiguity resolution does not work when both ϕ_{ref} and the counter would be transitioning at the same time. Here, the delay τ plays an important role. Its value should be large enough such that the transitions in the reference phase and the counter states are sufficiently spaced apart. However, the delay should also remain smaller than half the VCO period to avoid that counter transitions start to coincide again with the next edge in ϕ_{ref}. In order to verify the operation under these circumstances, we discuss now sample instant T_5 on the right side of Fig. 3. Here, the situation is comparable to what happened at T_1: none of the signals are ill defined. However, because τ is assumed to be much larger, the ϕ_{ref} sample is now zero, i.e., the expected parity is now different from the counter parity. As a result, the counter will be incremented, leading to an extra contribution N in (1). However, note that the fine counter has just wrapped around, and hence made a step N downwards. Both these contributions cancel completely, leading in the end again to the expected output value.

Having explained the range of allowed delays for which the compensation method works, we can see that the assumption being made w.r.t. the equalization can to a large extent be relaxed. Indeed, from the description of the ambiguity resolution it is clear that only the last delay is of importance. Therefore, code-dependent delays do not pose a problem, as long as all delays stay within the safe limits discussed above. The situation is also very similar for other sources of delay variability, such as process, supply and temperature (PVT) variations. Ultimately, the maximum delay τ_{max} that can occur sets an upper bound on the VCO frequency that can be reliably processed: $f_{\text{VCO}} < 1/(2\tau_{\text{max}})$.

Fig. 4 shows an efficient implementation of the decoding logic. As an example we show the 7-bit case. The decoder block receives the sampled Gray code $G_{6\ldots0}$ and the expected parity bit P (i.e., the sample of the reference phase) and outputs

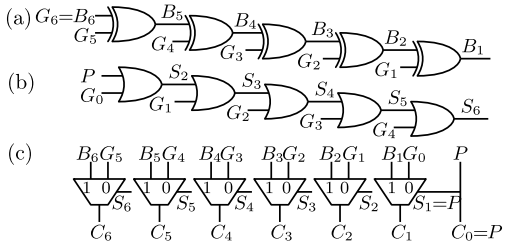
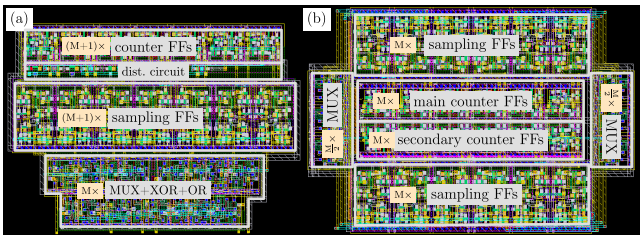


Fig. 4. Decoding logic consisting of Gray-to-binary conversion circuit and ambiguity resolution based on the expected parity P.



(c)

	#counter FFs	#sampling FFs	#MUX	#XOR	#OR	M=6 current [μA]	M=6 area [μm^2]
This work	M+1	M+1	M	M	M	**3.3**	**1102**
Perez '23 [9]	2M	2M	M	0	0	4.0	1394

Fig. 5. (a) layout of the new design, (b) layout of the reference design, (c) comparison table.

the corrected digital code $C_{6\ldots0}$. The circuit comprises (a) an XOR chain for Gray-to-binary conversion, (b) an OR chain for generating MUX selection signals, (c) MUX gates that select the correct output. Note the repetitive pattern of the circuit allows to easily generalize to other bit-lengths. Also, in general the length of the sequential paths does not pose problems because the logic operates at the fairly low sample frequency f_s.

IV. Comparative Designs & Simulation Results

In order to provide an in-depth comparison with prior art, an 11 bit coarse-fine ADC architecture is designed in a standard 180nm CMOS process consisting of a ring oscillator with 16 differential delay elements, a 7-bit version of the new Gray counter of Fig. 2 with as FF the dual of [19, Fig. 13c], the decoding logic of Fig. 4, and sense amplifiers (SAs) to sample the VCO and counter state. As a reference design a very similar structure is worked out, but based on a 6-bit prior-art double ripple counter and ambiguity resolution according to [9]. Note that, in order to make a fair comparison, the double-edge triggered Gray counter needs one bit extra to arrive at the same 11-bit range as the reference design. The layout of the new and the reference design is shown in Fig. 5a and Fig. 5b, respectively. After RC extraction of the layouts, simulations were performed. The testbench consists of a sinusoidal input signal being applied to the VCO, making the VCO frequency vary between 13 MHz and 63 MHz. The signal frequency is 201kHz and the sample frequency 1 MHz, resulting in 5 samples per input signal period that slowly change over time. Fig. 6a shows the differentiated output of the decoding logic, which represents the digitized input signal. Because the coarse count has a relatively large weight, errors in ambiguity resolution would be visible in this plot,

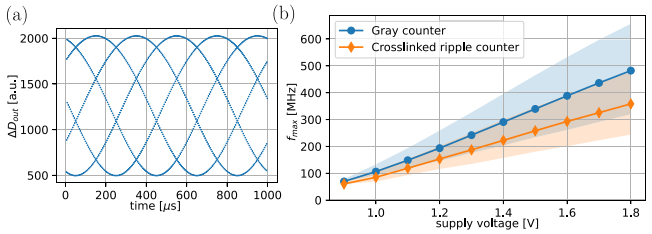


Fig. 6. (a) digital output of simulated coarse-fine ADC, (b) maximum counter frequency as a function of supply voltage.

which is not the case. As an extra check that all ambiguities were correctly resolved, the quantization error was numerical evaluation and scanned for outliers. The area of the layouts and the current consumption for the described simulation are reported in Fig. 5c. With an area of $1102\mu m^2$ versus $1394\mu m^2$, the new design is 21% smaller. At the same time, the current consumption obtained from the post-layout simulations show a reduction from $4.0\mu A$ for the reference design to $3.3\mu A$ for the new design, i.e., 17.5% lower.

In order to evaluate the impact of PVT variations, the maximum counter frequency f_{max} has also been determined at $-40°C$, $27°C$ and $+165°C$, and over the best and worst process corner, as a function of the supply voltage going from 0.9V up to the nominal 1.8V. For this, the VCO circuit was replaced by an input clock signal that switched between 0 and the simulated supply voltage. The frequency is ramped up from zero to maximally 750 MHz. Again results are from post-layout simulations. Fig. 6b shows the nominal results as well as upper and lower limits. The plot reveals that the new Gray counter also outperforms the reference design w.r.t. f_{max}. This can be attributed to the fact that the Gray counter has a more consistent delay, dominated by a single FF toggle event and the (equalized) delay of the pulse propagation network. In contrast, the cross-coupled ripple counter intermittently has long carry propagation chains where all FFs toggle sequentially. Above some input frequency, a systematic difference in propagation speed between the main and the secondary counter (due to unequal loading of the FFs because of the cross-coupling) leads to bit errors in the secondary counter MSB value.

In order to support more general conclusions, we summarized in Fig. 5c also the number of counter FFs, sampling FFs, and logic functions needed for the ambiguity resolution (MUX, XOR and OR functions) as a function of the number M of effective coarse counter bits. Upon closer examination of the table and the layouts, it becomes clear that the reduction in area of the new design can mostly be attributed to XOR and OR gates being smaller than FFs. Based on this observation, the area advantage of the new design is expected to hold for a broader range of bit lengths.

V. Conclusion

This brief has presented an asynchronous Gray counter that can advantageously be used as coarse counter in a VCO ADC. The counter is double-edge triggered and provides, together with the fine state, enough information to avoid ambiguity in the counter state. The core Gray counter structure is shown to be about 21% smaller in size and 17.5% more power efficient compared to the already excellent performance of a double-sampling binary ripple counter approach. The area advantage is expected to hold for a broader range of bit lengths.

References

[1] G. G. Gielen, L. Hernandez, and P. Rombouts, "Time-encoding analog-to-digital converters: Bridging the analog gap to advanced digital CMOS—Part 1: Basic principles," *IEEE Solid-State Circuits Mag.*, vol. 12, no. 2, pp. 47–55, Jun. 2020.

[2] G. G. Gielen, L. Hernandez, and P. Rombouts, "Time-encoding analog-to-digital converters: Bridging the analog gap to advanced digital CMOS—Part 2: Architectures and circuits," *IEEE Solid-State Circuits Mag.*, vol. 12, no. 3, pp. 18–27, Aug. 2020.

[3] G. Gielen, L. Hernandez-Corporales, and P. Rombouts, *Time-Encoding VCO-ADCs for Integrated Systems-on-Chip: Principles, Architectures and Circuits*. Cham, Switzerland: Springer, 2022. [Online]. Available: http://doi.org/10.1007/978-3-030-88067-5

[4] J. Borgmans, R. Riem, and P. Rombouts, "The analog behavior of pseudo digital ring oscillators used in VCO ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 68, no. 7, pp. 2827–2840, Jul. 2021.

[5] J. Borgmans, E. Sacco, P. Rombouts, and G. Gielen, "Methodology for readout and ring oscillator optimization toward energy-efficient VCO-based ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 69, no. 3, pp. 985–998, Mar. 2022.

[6] J. Daniels, W. Dehaene, and M. Steyaert, "All-digital differential VCO-based A/D conversion," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2010, pp. 1085–1088.

[7] A. Quintero et al., "A coarse-fine VCO-ADC for MEMS microphones with sampling Synchronization by data scrambling," *IEEE Solid-State Circuits Lett.*, vol. 3, pp. 29–32, 2020.

[8] C. Perez, A. Quintero, P. Amaral, A. Wiesbauer, and L. Hernandez, "A 73dB-a audio VCO-ADC based on a maximum length sequence generator in 130nm CMOS," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 68, no. 10, pp. 3194–3198, Oct. 2021.

[9] C. Perez et al., "A VCO-based ADC with direct connection to a microphone MEMS, 80-dB peak SNDR and 438-μW power consumption," *IEEE Sensors J.*, vol. 23, no. 8, pp. 8466–8477, Apr. 2023.

[10] V. Nguyen, F. Schembari, and R. B. Staszewski, "Exploring speed maximization of frequency-to-digital conversion for ultra-low-voltage VCO-based ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 70, no. 3, pp. 1043–1056, Mar. 2023.

[11] T. Okazawa and I. Akita, "A robust and low-power synchronization technique of coarse-and-fine conversion parts in ring-oscillator-based time-to-digital converters," in *Proc. 24th IEEE Int. Conf. Electron., Circuits Syst. (ICECS)*, 2017, pp. 239–242.

[12] J. Daniels, W. Dehaene, M. Steyaert, and A. Wiesbauer, "A 0.02mm2 65nm CMOS 30MHz BW all-digital differential VCO-based ADC with 64dB SNDR," in *Proc. Symp. VLSI Circuits*, 2010, pp. 155–156.

[13] M. Baert and W. Dehaene, "A 5-GS/s 7.2-ENOB time-interleaved VCO-based ADC achieving 30.5 fJ/cs," *IEEE J. Solid-State Circuits*, vol. 55, no. 6, pp. 1577–1587, Jun. 2020.

[14] K.-C. Choi, S.-W. Lee, B.-C. Lee, and W.-Y. Choi, "A time-to-digital converter based on a multiphase reference clock and a binary counter with a novel sampling error corrector," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 3, pp. 143–147, Mar. 2012.

[15] J. Huang and P. P. Mercier, "A 112-dB SFDR 89-dB SNDR VCO-based sensor front-end enabled by background-calibrated differential pulse code modulation," *IEEE J. Solid-State Circuits*, vol. 56, no. 4, pp. 1046–1057, Apr. 2021.

[16] V. Unnikrishnan and M. Vesterbacka, "Mitigation of sampling errors in VCO-based ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 7, pp. 1730–1739, Jul. 2017.

[17] V. Medina, R. Garvi, E. Gutierrez, S. Paton, and L. Hernandez, "A gray-encoded ring oscillator for efficient frequency-to-digital conversion in VCO-based ADCs," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 70, no. 3, pp. 870–874, Mar. 2023.

[18] V. Unnikrishnan and M. Vesterbacka, "Time-mode analog-to-digital conversion using standard cells," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 12, pp. 3348–3357, Dec. 2014.

[19] J. Yuan and C. Svensson, "High-speed CMOS circuit technique," *IEEE J. Solid-State Circuits*, vol. 24, no. 1, pp. 62–70, Feb. 1989.