

Article

A 1-to-8 Programmable Clock Divider with a 50% Duty Cycle and Low Jitter for ADCs

Yuxing Zhang , Yanhan Gu, Li Zeng , Ming Wang , Rui Yin and Zhangwen Tang *

State Key Laboratory of ASIC and System, Fudan University, Shanghai 200433, China;
21112020122@m.fudan.edu.cn (Y.Z.); 23112020065@m.fudan.edu.cn (Y.G.); 19112020034@m.fudan.edu.cn (L.Z.);
21112020110@m.fudan.edu.cn (M.W.); yinrui@fudan.edu.cn (R.Y.)

* Correspondence: zwtang@fudan.edu.cn

Abstract

In this paper, a programmable clock divider with a 50% duty cycle and low jitter for analog-to-digital converters (ADCs) is presented. The proposed divider, using a 180 nm CMOS process, can handle an input clock range from 40 MHz to 1 GHz. It supports integer division ratios from 1 to 8, with programmable phase shifting corresponding to each division ratio. To address the issue of the non-50% duty cycle with odd division ratios, this design combines edge-triggered and level-triggered flip-flops to simultaneously perform logical operations during the falling edge and high-level periods of the input clock, ensuring that the output clock maintains an accurate 50% duty cycle under all integer division ratios. Moreover, a clock-edge synthesis technique is adopted: three dividing timing signals are first generated to extract the pure edges of the input clock, which suppresses the excess jitter introduced by conventional multi-stage flip-flop chains. Simulation results show that, with a 1 GHz input clock, the RMS value of the jitter is less than 85.12 fs under the worst process corner, and the power consumption is less than 3.037 mW.

Keywords: frequency division; phase shifting; 50% duty cycle; clock-edge synthesis technique; low jitter

1. Introduction

In modern mixed-signal systems, high-speed and high-precision ADCs are one of the core modules that determine the performance of the system. The clock divider, as a key circuit of the ADC clock management unit, needs to have flexible frequency division and phase-shift functions to convert a high-frequency input clock into a lower sampling clock suitable for ADC operation. If the signal output by the divider has excessive jitter or duty-cycle distortion, it will directly deteriorate the dynamic performance of the ADC, such as the signal-to-noise ratio (SNR).

In ADCs, clock dividers are commonly implemented using Current-Mode Logic (CML), True Single-Phase Clock (TSPC) logic, and CMOS logic. Among these, CML dividers can handle high-frequency clock signals up to tens of GHz, but their power consumption is relatively high [1–3]. The working frequency and power consumption of TSPC logic dividers [4–6] lie between those of CML and CMOS logic dividers. In contrast, at lower frequencies, CMOS logic dividers [7–9] are the best choice in terms of power. Regarding jitter performance, CML frequency dividers exhibit the lowest jitter due to their differential operation and immunity to common-mode noise; for instance, a 40 GHz CML divider achieves an RMS jitter of only 0.8 ps [10]. TSPC dividers offer



Received: 30 January 2026

Revised: 27 February 2026

Accepted: 3 March 2026

Published: 4 March 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

moderate jitter performance [11,12]. In contrast, CMOS logic dividers suffer from higher jitter accumulation due to larger gate delays in multi-stage flip-flop chains [13].

The most efficient way to implement a clock divider in a given CMOS technology is often to synthesize it from Verilog HDL using standard cells [14–16]. However, this standard cell-based design approach exhibits several limitations compared to a full-custom design. Firstly, it supports only a limited input clock frequency. Secondly, its division logic, often based on synchronous counters, fails to generate a precise 50% duty-cycle output when odd division ratios are required. Moreover, significant additional jitter is introduced due to signal propagation through multi-stage flip-flop chains. Finally, it generally occupies a large circuit area. Therefore, designing a high-performance clock divider with low jitter, an accurate 50% duty cycle, and flexible frequency/phase control is essential for improving the overall performance of ADC systems.

This paper is organized as follows: Section 2 investigates and analyzes the key techniques for achieving a 50% duty cycle for odd division ratios and reducing output clock jitter. Section 3 introduces the specific circuits, mainly including the core division logic based on a counter and phase shifter, the dual flip-flop combination technique, and the clock-edge synthesis technique. Section 4 presents the circuit simulation results obtained under all process corners. Finally, Section 5 concludes the work presented in this paper.

2. Analysis of the Proposed Techniques

2.1. A 50% Duty Cycle for Odd Division Ratios

The sampling of an ADC usually occurs at the rising or falling edge of the clock signal. When the duty cycle of the clock signal deviates from 50%, the sampling points will shift, and additional harmonic distortion will be introduced, significantly degrading the Spurious-Free Dynamic Range (SFDR) of the SNR [17–19]. This effect is particularly severe when using Nyquist sampling or high-frequency under-sampling.

Ref. [20] proposed a configurable integer-N clock divider architecture with synchronous reset; the duty cycle of the output is 50% when using even division ratios and approaches 50% when using odd division ratios. Ref. [5] proposed a synchronously started divide-by-five clock divider; the output duty cycle is 40%. Ref. [21] designed a multi-frequency generator; they achieved an output of 50% through an additional duty-cycle correction circuit. Ref. [22] proposed a method of adjusting an equal number of rising edges and falling edges to achieve a 50% duty cycle, even in odd division ratios. While addressing the duty-cycle issue, the power consumption caused by the additional circuit also increases. Achieving a 50% duty cycle in odd division (e.g., division by 7) requires the output to toggle after half-integer input clock cycles (e.g., 3.5 cycles). However, conventional flip-flops can only toggle at integer multiples of the input clock period. To overcome this limitation, the proposed design combines both edge-triggered and level-triggered flip-flops. Specifically, it generates triggering actions on the falling edge and during the high-level period of the input clock, creating a half-cycle offset between toggling events. This technique enables precise 50% duty cycle generation for any odd division ratio without additional circuitry or power consumption.

Usually, frequency dividers based on synchronous counters and logical operations can easily achieve a 50% duty cycle when dividing by even ratios, but the duty cycle for odd division ratios can only approach 50%. At this time, the high level of the output will be one input clock cycle longer than the low level. Suppose that, if we can divide the extra clock cycle of the high level by half for the low level, then odd division ratios can also achieve a 50% duty cycle. As shown in Figure 1, both I1 and I2 employ complementary clock inputs (CK/CKN for I1; G/GN for I2), modified from conventional single-clock flip-flops. I1 operates as an edge-triggered flip-flop: on the rising edge of CK (the falling edge of CKN),

Q follows D; otherwise, Q retains its previous state. I2 operates as a level-triggered flip-flop: during the high level of G (the low level of GN), Q follows D; otherwise, Q holds. To realize this scheme, the master input clock ck_{in} is externally split into two complementary signals, ck and ckn , rather than generating the inverted clock internally within each flip-flop. This external generation simplifies the flip-flop design and enhances the overall operating speed. All flip-flops presented in this work adopt this configuration. As a result, I1 is triggered at the rising edge of ckn (corresponding to the falling edge of ck_{in}), while I2 is transparent during the high period of ck (corresponding to the high period of ck_{in}). Consequently, the two triggering instances are spaced exactly half a clock cycle ($T/2$) apart, as depicted by waveforms Q1 and Q2 in Figure 1.

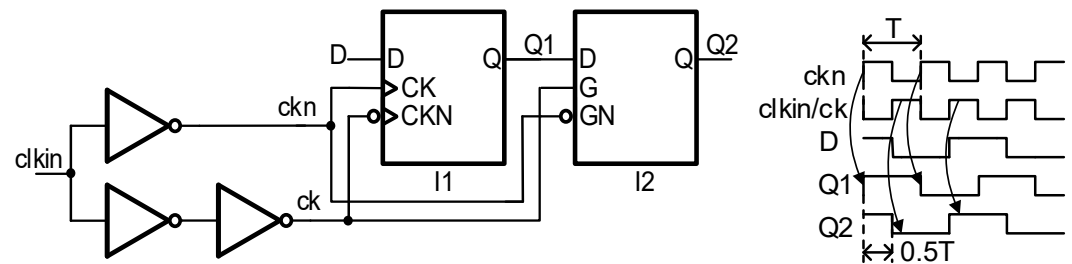


Figure 1. A combination of edge-triggered and level-triggered flip-flops.

2.2. The Clock-Edge Synthesis Technique

The jitter of a clock signal is a key indicator for assessing its time-domain stability, referring to the random deviation of the clock edge from its ideal position in time [23,24]. In ADCs, the degree of deterioration of the SNR caused by clock jitter can be approximately estimated using the following formula [25]:

$$SNR_{jitter}(dB) \approx -20 \log_{10}(2\pi f_{in} t_j) \quad (1)$$

where f_{in} is the frequency of the analog input signal in Hertz (Hz), and t_j is the jitter of the clock signal in seconds (s). It can be seen that, the higher the frequency of the analog input signal, or the greater the jitter of the clock signal, the worse the SNR of the ADC [26–28]. Therefore, for high-speed and high-precision ADCs, a sampling clock with extremely low jitter is of crucial importance.

Figure 2a shows the structure of a conventional clock divider, in which the divider can be synthesized from Verilog. The rising edge and the falling edge of the output $clkout$ align with the rising edge of the input ck_{in} , and the additional jitter of the $clkout$ rising and falling edges is the jitter generated by the entire divider chain, usually consisting of multiple flip-flops. Refs. [16,22,29,30] minimize clock jitter by reducing and optimizing the length of the transmission chain.

Figure 2b presents the divider structure proposed in this paper. It employs the clock-edge synthesis technique commonly used in time-interleaved ADCs for generating multiphase clocks [31,32]. To minimize additional jitter caused by the multi-stage flip-flop chains, the output clock is not generated directly from the divider chain. Instead, the division circuit first generates three dividing timing signals, $out1$, $out2$, and $out3$, which are then processed through NAND logic operations with the input clock ck_{in} to produce the output $clkout$. Furthermore, $out1$ and $out2$ are used to capture clean rising/falling edges of the input clock, while $out3$ controls the output toggling instant to achieve a precise 50% duty cycle. As shown in the waveforms in the upper right corner of Figure 2b, the rising/falling edge of the $clkout$ aligns with the falling/rising edge of the ck_{in} , and the additional jitter of the $clkout$ rising and falling edge is the jitter of the NAND logic gates. The advantage of this approach is that the additional jitter introduced by the logic gates

is much smaller than that introduced by multiple flip-flops. Furthermore, we can reduce the jitter of the output clock by increasing the width-to-length ratio (W/L) of the logic gate transistors.

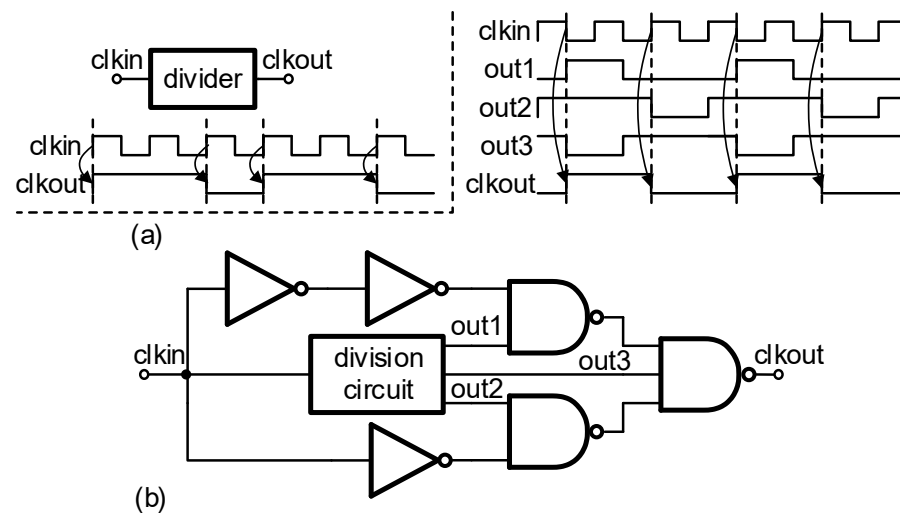


Figure 2. (a) The conventional clock divider. (b) The proposed clock divider.

3. Circuit Design

3.1. Counter and Phase Shifter

The circuit of a 3-bit binary synchronous counter is shown in Figure 3. The 3-bit division ratio control signal $divr <2:0>$ implements the frequency division function by controlling the counting states of the counter. When $divr <2:0> = N$ (where $0 \leq N \leq 7$), the counter will cycle through $N + 1$ states sequentially and achieve $N + 1$ division. Furthermore, $cntr <2:0>$ transmits the counter's state to the subsequent phase shifter, which then cooperates to generate a phase-shifted carry pulse signal.

Figure 4 shows the circuit of the phase shifter. The signal $rstn$ serves as an asynchronous reset input (active low). The three XNOR gates (I12, I13, and I14) detect whether the 3-bit phase control signal $padj <2:0>$ has changed. If a change is detected, the signal $alterph$ generates a high-level pulse at the third rising edge of ckn to reselect $padj <2:0>$. Figure 5a,b show the waveforms for the case where $divr <2:0> = 010$ (division ratio = 3) and $padj <2:0> = 000$ (phase shift = 0) or 001 (phase shift = 1). When the division ratio is 3, the counter's output state $cntr <2:0>$ cycles from 000 to 010 sequentially: If $padj <2:0> = 000$, the three XNOR gates (I16, I17, and I18) output 0 when and only when $cntr <2:0> = 000$; consequently, the signal co generates a high-level pulse at the immediately following rising edge of ckn . If $padj <2:0> = 001$, the three XNOR gates output 0 when and only when $cntr <2:0> = 001$; the signal co generates a high-level pulse at the second rising edge of ckn thereafter. Therefore, the signal co is a carry pulse that encapsulates both frequency division and phase-shift information. Since the circuit operates fully synchronously, with output transitions occurring only at the clock edge, and all logic gates have propagation delays significantly shorter than the clock period, any combinational logic settles well before the next active clock edge. Therefore, the phase-shift accuracy is not compromised.

To ensure correct phase-shift timing, an additional circuit must be designed at the output stage of the phase shifter to suppress the first potential incorrect high-level pulse on the signal co . In Figure 4, when the division ratio is 5, the counter's output state $cntr <2:0>$ cycles from 000 to 100 sequentially. And $padj <2:0>$ needs four rising edges of ckn before it reaches I16, I17, and I18 for comparison with $cntr <2:0>$. At this point, $cntr <2:0>$ has changed to 011. If $padj <2:0>$ is 000 (phase shift = 0), 001 (phase shift = 1), or 010 (phase shift = 2), co will output a high-level pulse at the 7th, 8th, and 9th rising edges of ckn .

However, if $\text{padj} \langle 2:0 \rangle$ is 011 (phase shift = 3), co will output a high-level pulse at the 5th rising edge of ck , which does not match the expected phase-shift timing. The modified circuit and waveforms are shown in Figure 6a,b. Here, I3 generates an inverted pulse of the signal alterph (Q3) at the 5th rising edge of ckn , thereby preventing the incorrect pulse of co from passing through I5 and I6. Thereafter, the output of I3 remains high, allowing co to pass normally. The signal co-delay is a version of co that is inverted and delayed by one ckin cycle; it masks the erroneous pulse from co at the 5th ckn rising edge and instead transmits the correct pulse occurring at the 10th ckn rising edge, thereby guaranteeing the correctness of the phase-shift timing.

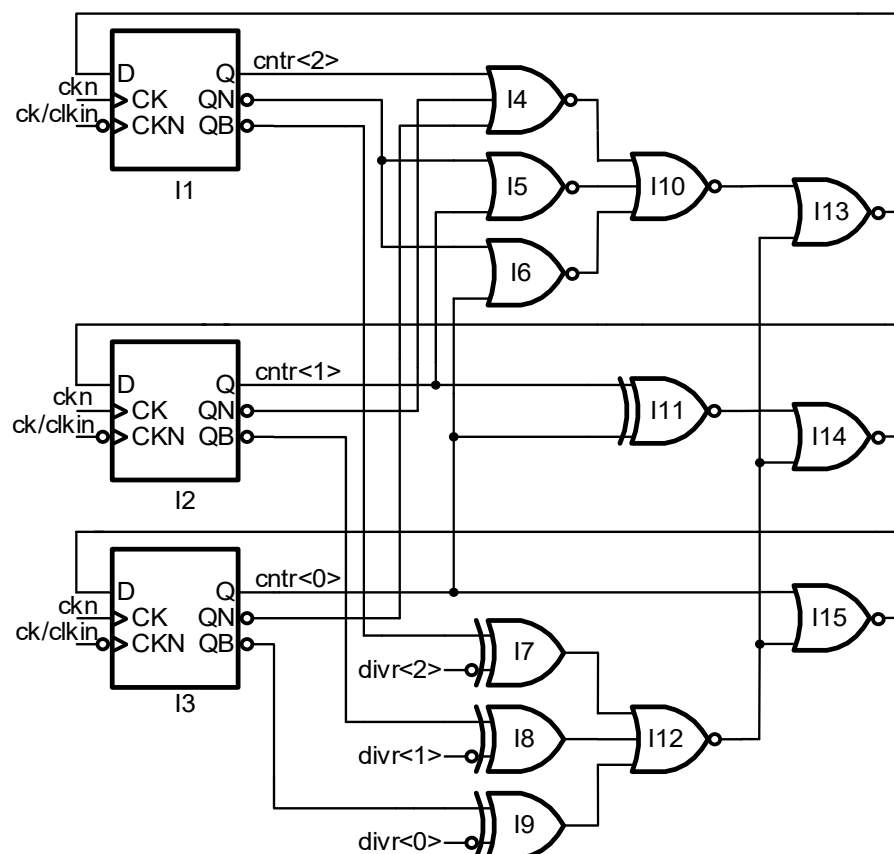


Figure 3. A 3-bit binary synchronous counter.

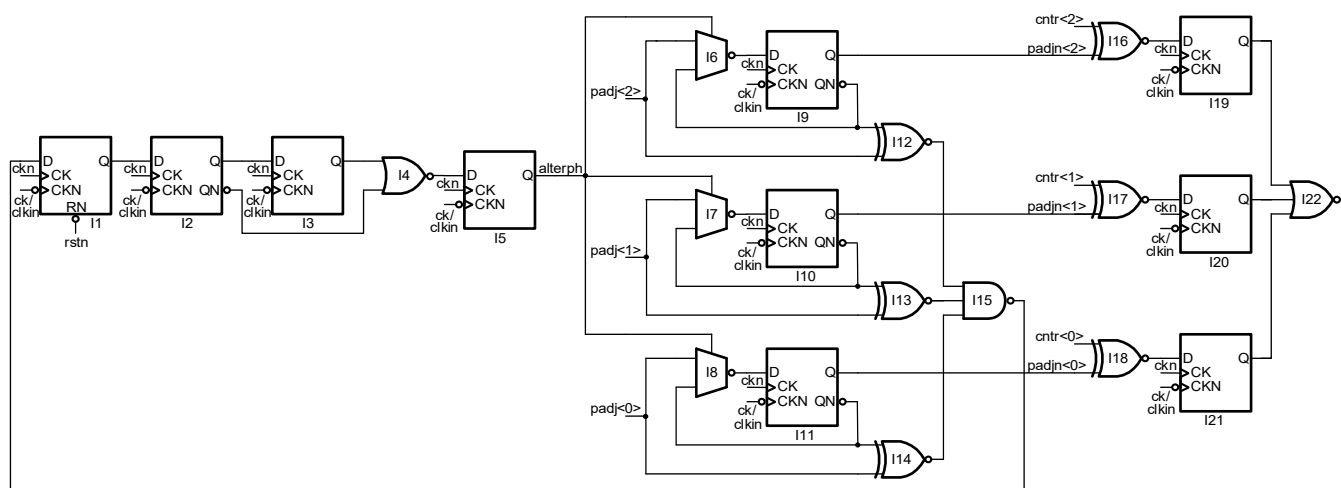


Figure 4. Phase shifter.

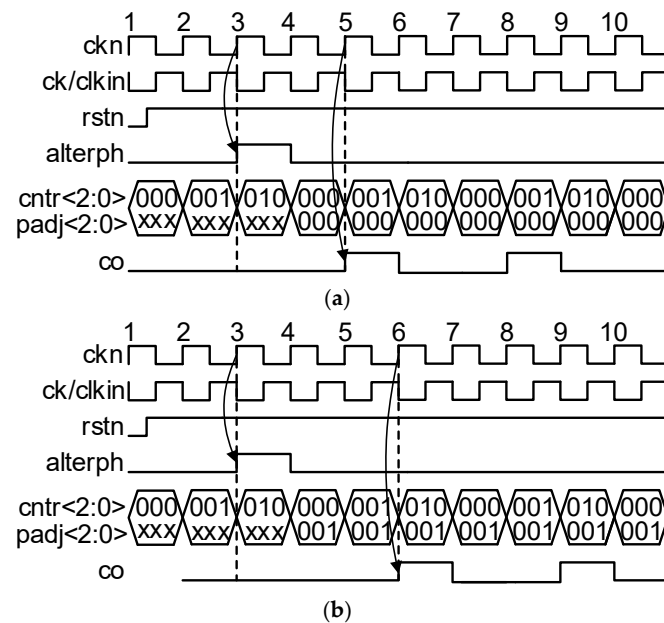


Figure 5. Division by 3. (a) Phase shift = 0. (b) Phase shift = 1.

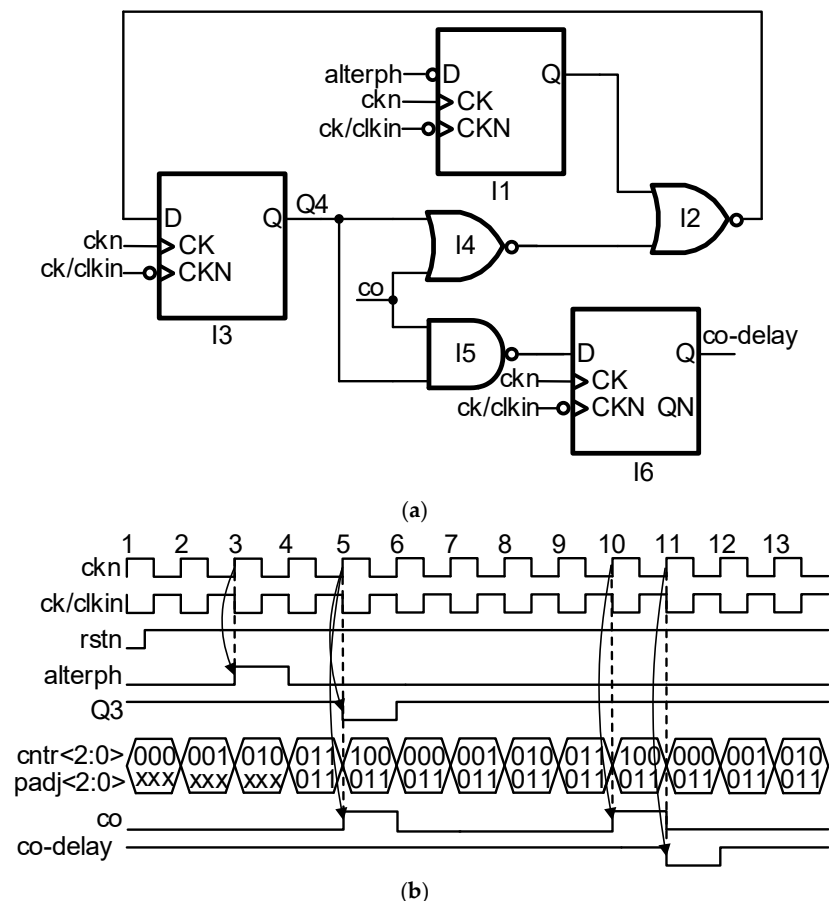


Figure 6. (a) The modified circuit. (b) Waveforms of the circuit.

3.2. Duty-Cycle Adjustment and Clock-Edge Synthesis

The carry pulse signal co-delay generated by the combination of the above counter and phase shifter has accomplished the functions of frequency division and phase shifting, but its duty cycle is close to 90%, which is not suitable for use by the ADC. The next step involves edge-triggered flip-flops with level-triggered flip-flops to first address the issue

of the non-50% duty cycle during odd division ratios. Subsequently, the clean rising and falling edges of the input clock signal will be extracted to output a low-jitter clock signal.

First, using the signal co-delay, two signals, reg1 and reg2, with duty cycles close to 50% are generated based on the principle of a shift register [33–35]. Figure 7 shows the circuit of the multi-channel selective shift register, where the co-delay serves as the control terminals for the data selectors. The signal co-delay is generated by the phase shifter modified circuit and is updated at the falling edge of ck/clkin. When co-delay is low, the combinational logic operation results of the division ratio control signal $\text{divr} < 2:0 >$ are loaded into the shift register on the next clock edge; when co-delay is high, the register shifts the previously loaded data. The co-delay's low pulse arrives at the select terminal of the multiplexer before the next falling edge of ck/clkin, thanks to the clk-to-q delay of the phase shifter flip-flops. This provides a stable select signal for the multi-channel shift register, ensuring proper data loading and shifting without timing violations. Figure 8a,b display the waveforms of the circuit for division ratios of 5 and 8. It can be seen that, within one cycle of reg1, if the division ratio is odd, the high level has one more clk_{in} period than the low level, and the duty cycle is approximately 50%; if the division ratio is even, the duration of the high and low levels is equal, and the duty cycle is 50%. Furthermore, reg2 is the signal obtained by delaying reg1 by one clk_{in} cycle and inverting it, which is used for subsequent circuits.

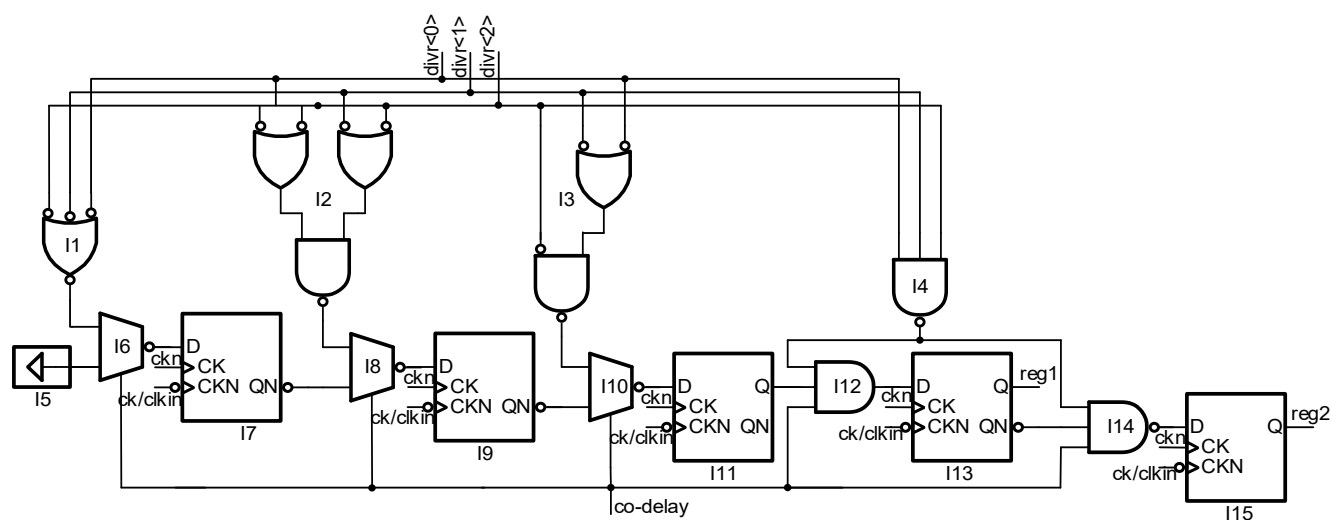


Figure 7. The multi-channel selective shift register.

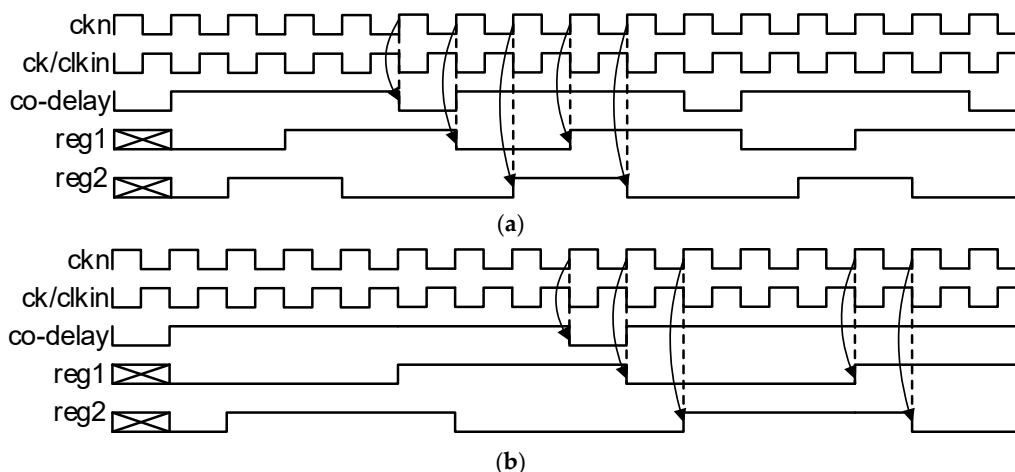


Figure 8. (a) Division by 5. (b) Division by 8.

The signals generated in all the above circuits are based on the edge-triggered flip-flops. Next, level-triggers will be applied. The purpose is to make the signals differ by half-integer multiples of the clk_{in} clock cycle so that, when using an odd division ratio, the extra high level of the signal $reg1$ can be evenly divided, thereby achieving a 50% duty cycle. The circuit is shown in Figure 9: I5, I6, and I7 are level-triggered flip-flops, $reg1$ and $reg2$ are based on edge-triggered flip-flops, and the signal $divr<0>$ indicates whether the division ratio is odd or even (0 for odd division and 1 for even division). After logical operations, three dividing timing signals, $out1$, $out2$, and $out3$, are generated for use in extracting the pure edge of the input clock signal in the subsequent stage. Figure 10 shows the timing relationship of the input and output signals when the frequency is divided by 5 and 8. It can be observed that the timing of $out2$ and $out3$ differs from that of $out1$ by half-integer multiples of the clk_{in} clock cycle.

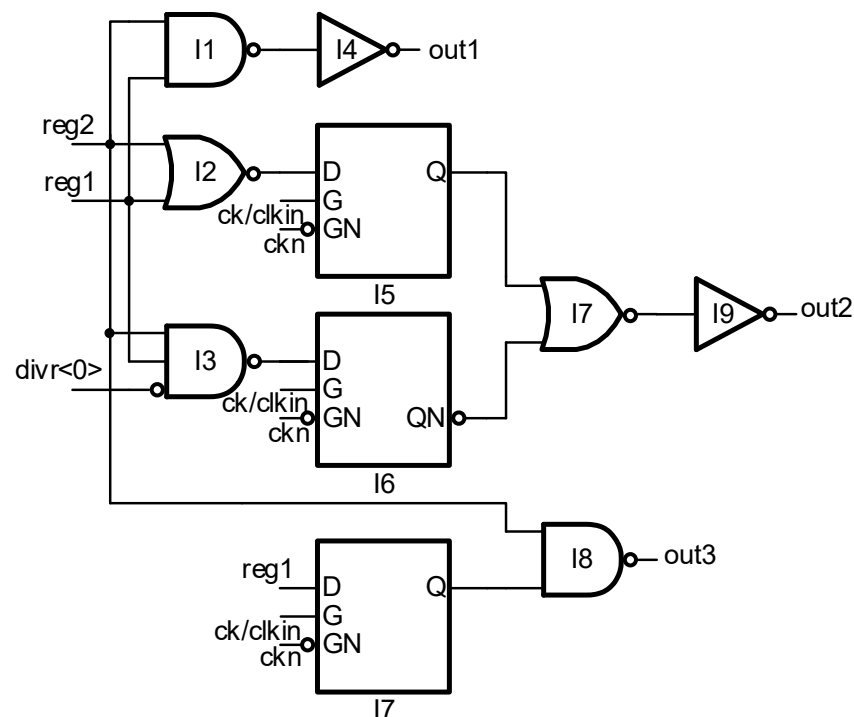


Figure 9. The generation of three dividing timing signals: $out1$, $out2$, and $out3$.

The clock-edge synthesis circuit is shown in Figure 11: $out1$, $out2$, and $out3$ act as three ordered gates controlling the transmission of the input clock clk_{in} . The additional jitter of the output clock clk_{out} is caused only by logic gates I1 to I6, so we can increase their size as much as possible to reduce the jitter. Figure 12a,b show the waveforms of $out1$, $out2$, and $out3$, extracting the rising and falling edges of the input clock clk_{in} and generating the output clock clk_{out} when the division ratios are 5 and 8, and the delays introduced by $out1$, $out2$, and $out3$ during the generation process relative to the input clock signal are considered. It can be seen that, if the additional jitter introduced by logic gates I1 to I6 is ignored, then, for odd division ratios, the rising edge and falling edge of the output clock clk_{out} are respectively the falling edge and rising edge of the input clock clk_{in} ; for even division ratios, the rising edge and falling edge of the output clock clk_{out} are both the falling edge of the input clock clk_{in} .

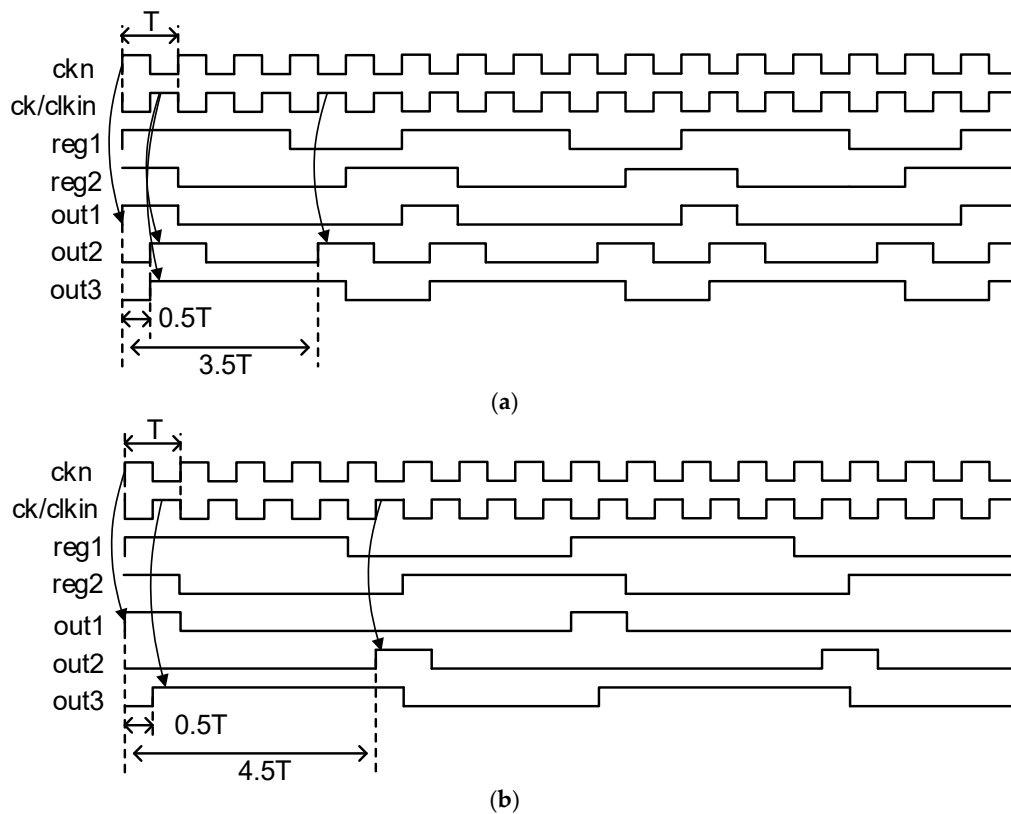


Figure 10. (a) Division by 5. (b) Division by 8.

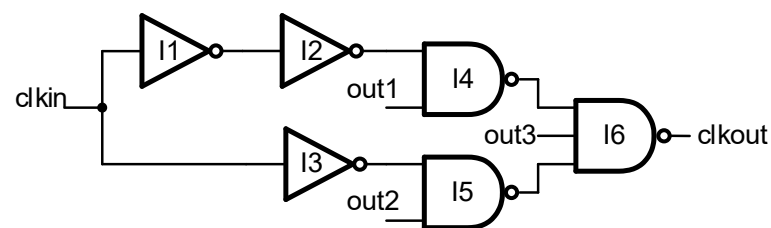


Figure 11. The clock-edge synthesis circuit.

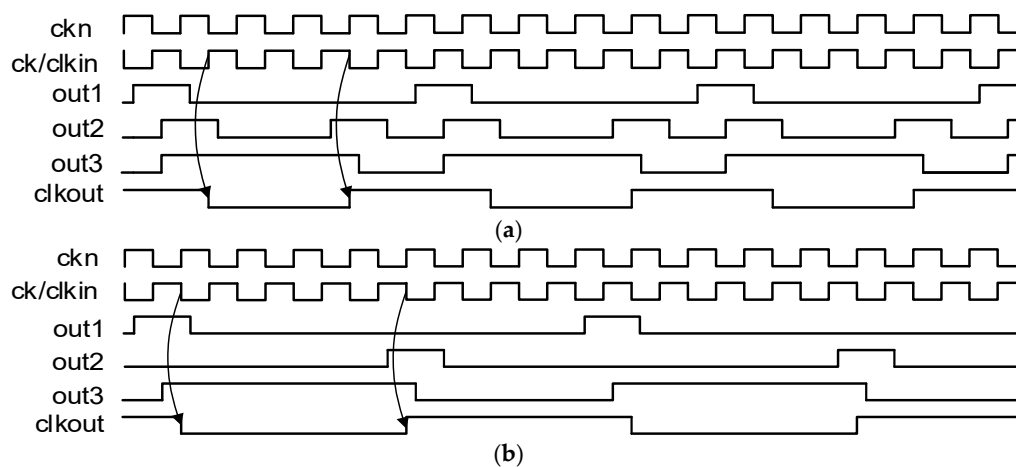


Figure 12. (a) Division by 5. (b) Division by 8.

4. Simulation Results

4.1. Frequency Division

The simulation results were obtained using circuit simulations performed in Cadence Virtuoso, with subsequent data processing and visualization conducted in MATLAB R2018b. Figures 13–15 present the simulation waveforms of the frequency division. In each case, the output clock frequency is 125 MHz. Specifically, Figure 13 shows the result with an input clock of 1000 MHz divided by 8; Figure 14 corresponds to an input of 600 MHz divided by 5; and Figure 15 demonstrates the case with a 125 MHz input and no division (or a division ratio of 1). As can be seen in these three figures, the proposed divider correctly performs the frequency division function for input frequencies up to 1000 MHz, and the output signal achieves a 50% duty cycle, even for odd division ratios.

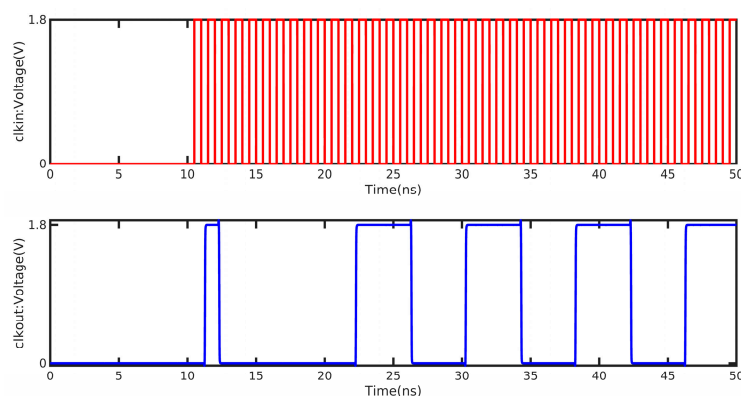


Figure 13. $f_{in} = 1000$ MHz, division by 8, output duty cycle of 50.5%.

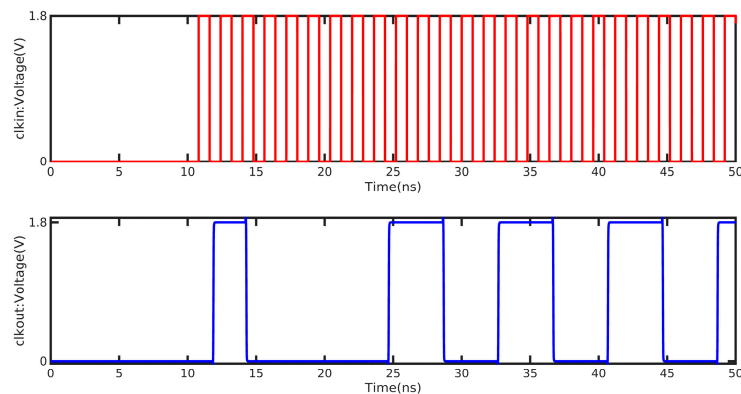


Figure 14. $f_{in} = 600$ MHz, division by 5, output duty cycle of 50.8%.

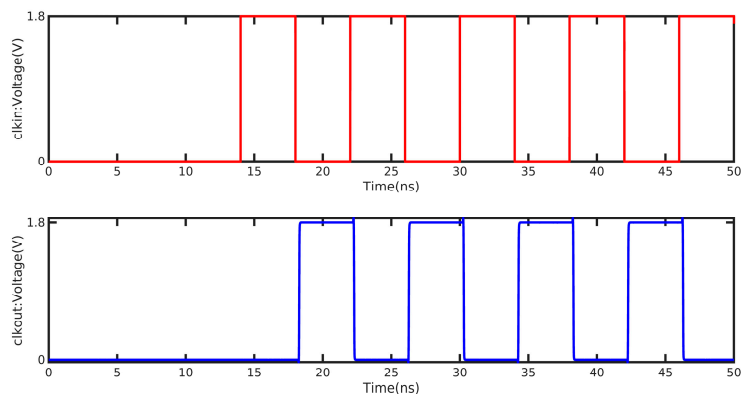


Figure 15. $f_{in} = 125$ MHz, division by 1, output duty cycle of 50.08%.

4.2. Phase Shifting

Figure 16 shows the phase-shift capability of the proposed divider for a 625 MHz input clock with a division ratio of 5. Five output waveforms are shown for phase-shift settings of 0 through 4, where each step corresponds to a delay of one input clock period. The progressive and uniform delay between successive waveforms confirms that the phase-shift function is operating normally.

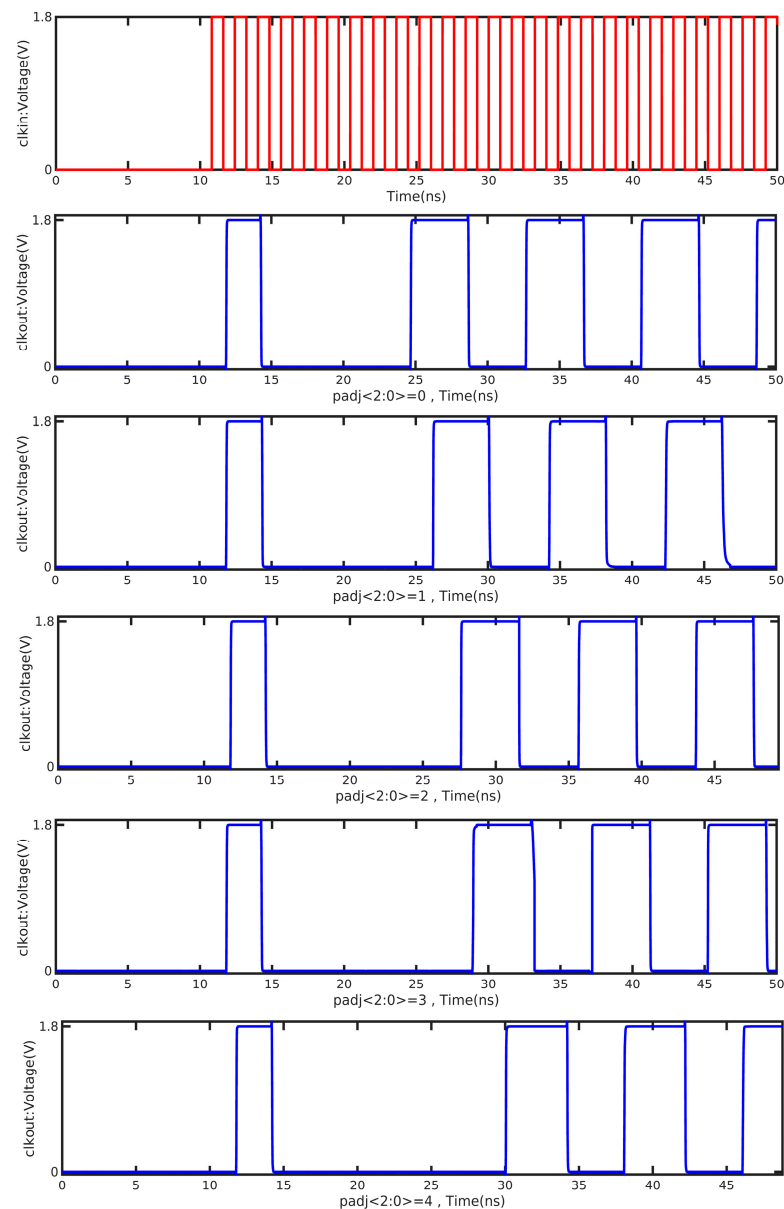


Figure 16. $f_{in} = 600$ MHz, division by 5, phase shift = 0, 1, 2, 3, 4.

4.3. Performance

Figures 17–19 show the relationship between the input clock frequency and the power consumption and the RMS jitter of the rising edge and falling edge of the output clock. Under a 1.8 V supply voltage, the performance was evaluated across five process corners: Typical (TT), Slow (SS), Fast (FF), Slow-NMOS-Fast-PMOS (SNFP), and Fast-NMOS-Slow-PMOS (FNFP). The results indicate that, across all process corners and frequency sweep conditions, the maximum power consumption is 3.037 mW. Regarding output timing jitter, the maximum value for the rising edge is 83.72 fs, while that for the falling edge is 85.12 fs.

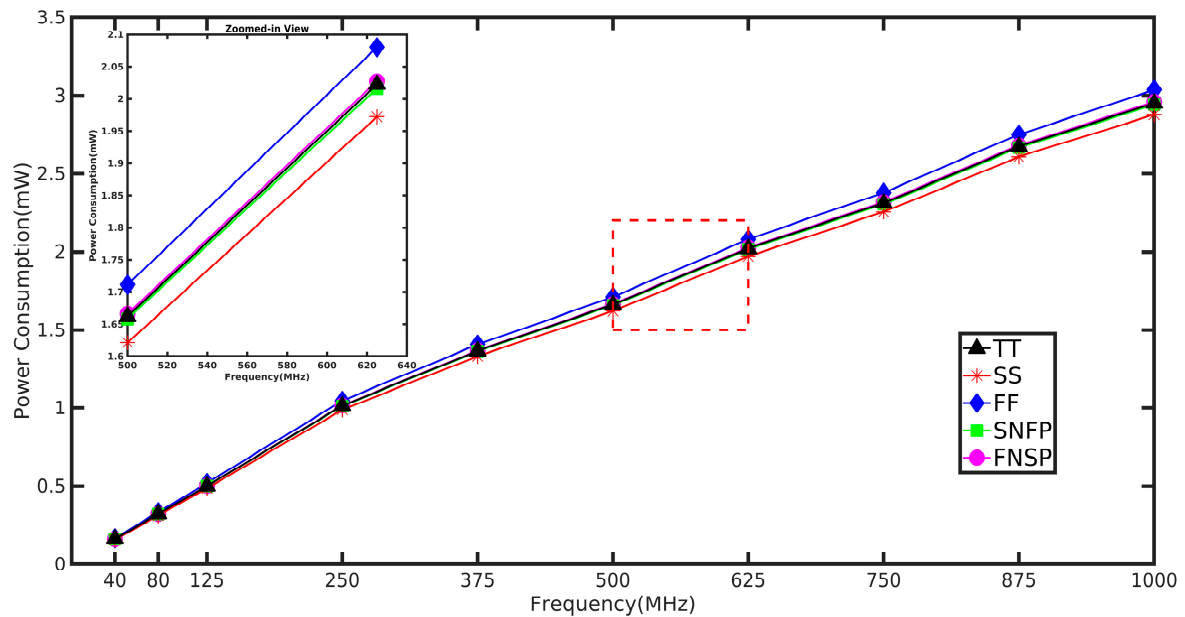


Figure 17. Power consumption (A magnified view of the region marked by the red box is provided in the top left corner).

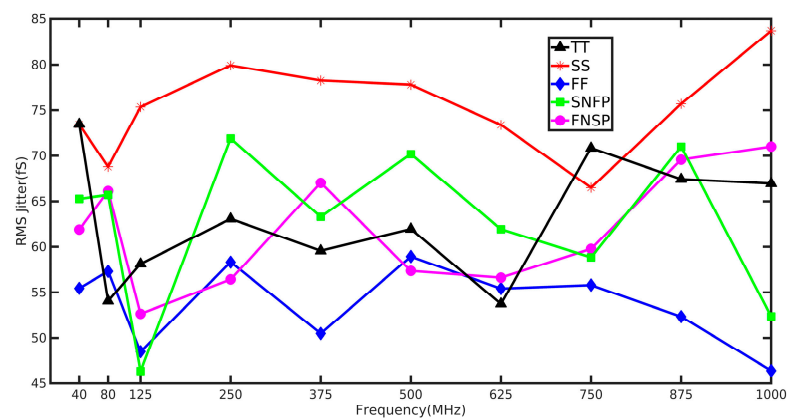


Figure 18. The RMS jitter of the output rising edge.

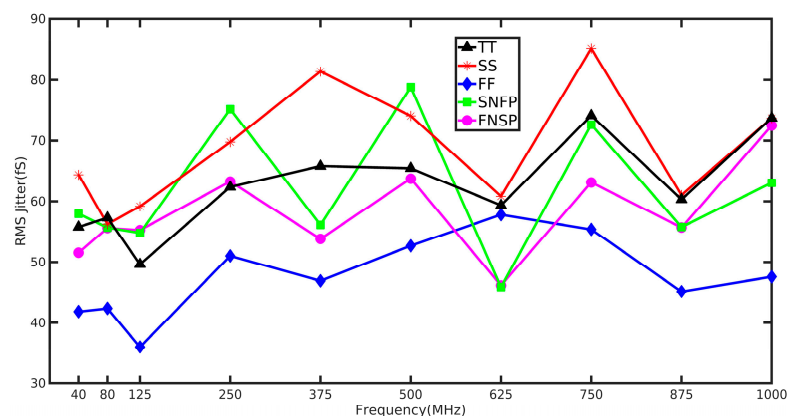


Figure 19. The RMS jitter of the output falling edge.

Table 1 compares the performance of the proposed divider with that of other designs. The results show that our design has significant advantages in terms of power consumption and jitter.

Table 1. Performance comparison.

	Ref. [29]	Ref. [36]	This Work
Process	90 nm CMOS	45 nm CMOS	180 nm CMOS
Maximum Input Frequency	4.7 GHz	60 GHz	1 GHz
Power	2.4 mW	9.6 mW	3.037 mW
RMS Jitter	1 ps	2.5 ps	85.12 fs
Area	0.0832 mm ²	0.01 mm ²	0.0138 mm ² *

* The area of this work is the post-layout core area.

5. Conclusions

This paper presents the design of a programmable frequency divider with an integer division ratio from 1 to 8, fabricated using a 180 nm CMOS process for high-speed ADC applications. An approach combining edge-triggered and level-triggered flip-flops is proposed to address the non-50% duty-cycle issue in odd division ratios, while a clock-edge synthesis technique is introduced to reduce signal jitter. Simulation results show that, under a 1.8 V supply voltage across various process corners, the maximum power consumption is 3.037 mW, and the RMS jitter of the output clock is at most 83.72 fs for the rising edge and 85.12 fs for the falling edge. The design demonstrates significant advantages in both power consumption and jitter performance. The design is optimized for ADC systems with a 125 MHz system clock, accepting input frequencies up to 1 GHz. Within this target frequency range, it demonstrates significant advantages in both power consumption and jitter performance for medium-speed data converter applications. Future work may explore higher-frequency operation through advanced process nodes or alternative topologies for ultra-high-speed applications.

Author Contributions: Data curation, Y.Z.; Formal analysis, Y.Z. and M.W.; Methodology, Y.Z.; Supervision, R.Y.; Validation, Y.Z. and L.Z.; Software, Y.Z. and Y.G.; Project administration, Z.T.; Writing—original draft, L.Z.; Writing—review and editing, Y.Z. and Y.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data used in this study can be requested from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

ADCs	Analog-to-Digital Converters
ADC	Analog-to-Digital Converter
SNR	Signal-to-Noise Ratio
CML	Current-Mode Logic
TSPC	True Single-Phase Clock
CMOS	Complementary Metal–Oxide–Semiconductor
PMOS	P-type Metal–Oxide–Semiconductor
NMOS	N-type Metal–Oxide–Semiconductor
HDL	Hardware Description Language
SFDR	Spurious-Free Dynamic Range
NAND	Not And
XNOR	Exclusive Not Or

W/L	Width-to-Length Ratio
RMS	Root Mean Squared
MATLAB	Matrix Laboratory
TT	Typical
SS	Slow
FF	Fast
SNFP	Slow-NMOS-Fast-PMOS
FNSP	Fast-NMOS-Slow-PMOS

References

- Patnaik, A.; Yoon, D. A 24–64 GHz Wideband Static CML Frequency Divider in a 90-Nm CMOS Technology. *IEEE Microw. Wirel. Technol. Lett.* **2025**, *35*, 71–74. [\[CrossRef\]](#)
- Rehman, B.K.; Sreeja, R.; Reddy, K.V.S.; Anuradha, P.; Nasar, M.A.; Basha, M. Design and Implementation of CML Frequency Divider Circuit Using 90nm Technology. In Proceedings of the 2024 IEEE International Conference of Electron Devices Society Kolkata Chapter (EDKCON), Kolkata, India, 30 November–1 December 2024; pp. 779–784. [\[CrossRef\]](#)
- Zhao, Y.; Qin, P.; Xue, Q. A 2.9 GHz–45.9 GHz CMOS CML Divider. In Proceedings of the 2025 International Conference on Microwave and Millimeter Wave Technology (ICMMT), Xi'an, China, 19–22 May 2025; pp. 1–3. [\[CrossRef\]](#)
- Krishna, M.V.; Jain, A.; Quadir, N.A.; Townsend, P.D.; Ossieur, P. A 1V 2mW 17GHz Multi-Modulus Frequency Divider Based on TSPC Logic Using 65nm CMOS. In Proceedings of the ESSCIRC 2014—40th European Solid State Circuits Conference (ESSCIRC), Venice Lido, Italy, 22–26 September 2014; pp. 431–434. [\[CrossRef\]](#)
- Probst, F.; Engelmann, A.; Weigel, R. A Synchronized 35 GHz Divide-by-5 TSPC Flip-Flop Clock Divider in 22 Nm FDSOI. In Proceedings of the 2023 Asia-Pacific Microwave Conference (APMC), Taipei, Taiwan, 5–8 December 2023; pp. 212–214. [\[CrossRef\]](#)
- Heydarzadeh, S.; Torkzadeh, P.; Pourmina, M. A 12GHz Programmable Fractional-n Frequency Divider with 0.18um CMOS Technology. In Proceedings of the 2013 5th Computer Science and Electronic Engineering Conference (CEEC), Colchester, UK, 17–18 September 2013; pp. 29–33. [\[CrossRef\]](#)
- Abdel-Hafeez, S.; Gordon-Ross, A. A Gigahertz Digital CMOS Divide-by-N Frequency Divider Based on a State Look-Ahead Structure. *Circuits Syst. Signal Process.* **2011**, *30*, 1549–1572. [\[CrossRef\]](#)
- Baluta, T.; Meyer, A.; Dossanov, A.; Kudabay, Y.; Bakhchova, L.; Issakov, V. A 7mW 22 GHz Frequency Divider Chain by 2048 Using CML, TSPC and CMOS Logic in 22FDX. In Proceedings of the 2025 IEEE 24th Topical Meeting on Silicon Monolithic Integrated Circuits in RF Systems (SiRF), San Juan, PR, USA, 19–22 January 2025; pp. 42–45. [\[CrossRef\]](#)
- Amirpour, M.; Akbari, S.; Hadidi, K.; Khoei, A. New All Digital Frequency Divider and Phase Shifter. In Proceedings of the 2014 22nd Iranian Conference on Electrical Engineering (ICEE), Tehran, Iran, 20–22 May 2014; pp. 205–208. [\[CrossRef\]](#)
- Kromer, C.; von Buren, G.; Sialm, G.; Morf, T.; Ellinger, F.; Jackel, H. A 40-GHz Static Frequency Divider with Quadrature Outputs in 80-Nm CMOS. *IEEE Microw. Wirel. Compon. Lett.* **2006**, *16*, 564–566. [\[CrossRef\]](#)
- Navickas, R. Design of Low Noise 10 GHz Divide by 16. .511 Frequency Divider. *Elektron. Elektrotech.* **2013**, *19*, 87–90. [\[CrossRef\]](#)
- Tseng, I.-W.; Wu, J.-M. An 18.7mW 10-GHz Phase-Locked Loop Circuit in 0.13-Mm CMOS. In Proceedings of the 2009 International Symposium on VLSI Design, Automation and Test, Hsinchu, Taiwan, 28–30 April 2009; pp. 227–230. [\[CrossRef\]](#)
- Cheung, H.; Singh, J. AFSM Circuit and Method for Low Jitter PLL CMOS Programmable Divider. U.S. Patent 7,683,679 B2, 23 March 2010.
- Wakhle, G.B.; Aggarwal, I.; Gaba, S. Synthesis and Implementation of UART Using VHDL Codes. In Proceedings of the 2012 International Symposium on Computer, Consumer and Control (IS3C), Taichung, Taiwan, 4–6 June 2012; pp. 1–3. [\[CrossRef\]](#)
- Mishra, K.K. *Advanced Chip Design, Practical Examples in Verilog*; CreateSpace Independent Publishing Platform: Scotts Valley, CA, USA, 2013.
- Azadmousavi, T.; Hadidi, K.; Khoei, A. Design of a Continuous Fractional Frequency Divider in 0.35μm CMOS Process. In Proceedings of the 2015 23rd Iranian Conference on Electrical Engineering (ICEE), Tehran, Iran, 10–14 May 2015; pp. 1133–1138. [\[CrossRef\]](#)
- Raja, I.; Banerjee, G.; Zeidan, M.A.; Abraham, J.A. A 0.1–3.5-GHz Duty-Cycle Measurement and Correction Technique in 130-Nm CMOS. *IEEE Trans. VLSI Syst.* **2016**, *24*, 1975–1983. [\[CrossRef\]](#)
- Dheena, D.R.; Vasanthi, A. Low Power and Area Efficient Duty Cycle Corrector with Sigma-Delta Analog to Digital Converter. In Proceedings of the 2013 International Conference on Information Communication and Embedded Systems (ICICES), Chennai, India, 21–22 February 2013; pp. 1188–1191. [\[CrossRef\]](#)
- Kim, J.; Lee, S.; Kim, B. All-Digital Fast-Locked Synchronous Duty-Cycle Corrector. *IEEE Trans. Circuits Syst. II Express Briefs* **2006**, *53*, 1363–1367.

20. Wagner, C.W.; Glaeser, G.; Kell, G.; Del Galdo, G. Every Clock Counts-41 GHz Wide-Range Integer-N Clock Divider. In Proceedings of the SMACD/PRIME 2021: International Conference on SMACD and 16th Conference on PRIME, Online, 19–22 July 2021; pp. 1–4.
21. N., D.R.; S., S.; Kavitha, K.R.; Vijayalakshmi, S.; Babu, B.M. Precision Clock: A Multi-Frequency Generator for Soc Application. In Proceedings of the 2025 Second International Conference on Intelligent Technologies for Sustainable Electric and Communications Systems (iTech SECOM), Coimbatore, India, 9–10 October 2025; pp. 1–5. [\[CrossRef\]](#)
22. Teja, J.N.; Vinodhini, M.; Kumar, P.S. Programmable Frequency Divider using Synchronous Binary Counter. In Proceedings of the 2025 6th International Conference for Emerging Technology (INCET), Belgaum, India, 23–25 May 2025; pp. 1–5. [\[CrossRef\]](#)
23. Jin, W.; Kim, M.-s.; Jo, C.-m.; Won, H.; Choi, K.-M. Automatic Clock Jitter Analysis Considering Clock Divider. In Proceedings of the 2009 International SoC Design Conference (ISOCC), Busan, Republic of Korea, 22–24 November 2009; pp. 41–44. [\[CrossRef\]](#)
24. Mo, X.; Wu, J.; Wary, N.; Carusone, T.C. Design Methodologies for Low-Jitter CMOS Clock Distribution. *IEEE Open J. Solid-State Circuits Soc.* **2021**, *1*, 94–103. [\[CrossRef\]](#)
25. Kester, W. (Ed.) *Data Conversion Handbook*; Elsevier/Newnes: Amsterdam, The Netherlands; Boston, MA, USA, 2005.
26. Brannon, B. *Sampled Systems and the Effects of Clock Phase Noise and Jitter*; Application Note AN-756; Analog Devices, Inc.: Wilmington, MA, USA, 2004.
27. Núñez, J.; Ginés, A.J.; Peralías, E.J.; Rueda, A. Low-Jitter Differential Clock Driver Circuits for High-Performance High-Resolution ADCs. In Proceedings of the 2015 Conference on Design of Circuits and Integrated Systems (DCIS), Estoril, Portugal, 25–27 November 2015; pp. 1–4. [\[CrossRef\]](#)
28. Zhang, M.; Yin, Y.; Deng, H.; Chen, H. Design of Low-Jitter Clock Duty Cycle Stabilizer in High-Performance Pipelined ADC. In Proceedings of the and Identification Anti-Counterfeiting, Security and Identification (ASID), Taipei, Taiwan, 24–26 August 2012; pp. 1–5. [\[CrossRef\]](#)
29. Ali, M.; Hegazi, E. A Multigigahertz Multimodulus Frequency Divider in 90-nm CMOS. *IEEE Trans. Circuits Syst. II Express Briefs* **2006**, *53*, 1333–1337. [\[CrossRef\]](#)
30. Zhong, N.; Zhang, R.; Shi, C.; Chen, J. A Bandwidth-Tracking Self-Biased 5-to-2800 MHz Low-Jitter Clock Generator in 55nm CMOS. In Proceedings of the 2018 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), Chengdu, China, 26–30 October 2018; pp. 57–60. [\[CrossRef\]](#)
31. Le Dortz, N.; Blanc, J.-P.; Simon, T.; Verhaeren, S.; Rouat, E.; Urdard, P.; Le Tual, S.; Goguet, D.; Lelandais-Perrault, C.; Benabes, P. 22.5 a 1.62GS/s Time-Interleaved SAR ADC with Digital Background Mismatch Calibration Achieving Interleaving Spurs below 70dBFS. In Proceedings of the 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), San Francisco, CA, USA, 9–13 February 2014; pp. 386–388. [\[CrossRef\]](#)
32. Guo, M.; Mao, J.; Sin, S.-W.; Wei, H.; Martins, R.P. A 1.6-GS/s 12.2-mW Seven-/Eight-Way Split Time-Interleaved SAR ADC Achieving 54.2-dB SNDR With Digital Background Timing Mismatch Calibration. *IEEE J. Solid-State Circuits* **2020**, *55*, 693–705. [\[CrossRef\]](#)
33. Weste, N.H.E.; Harris, D.M. *CMOS VLSI Design: A Circuits and Systems Perspective*, 4th ed.; Addison Wesley: Boston, MA, USA, 2011.
34. Smith, M.J.S. *Application-Specific Integrated Circuits*; Addison-Wesley: Reading, MA, USA, 1997.
35. Mano, M.M.; Ciletti, M.D. *Digital Design: With an Introduction to the Verilog HDL, VHDL, and SystemVerilog*, 6th ed.; Pearson: Hoboken, NJ, USA, 2018.
36. Rehman, S.U.; Ferchichi, A.; Khafaji, M.M.; Carta, C.; Ellinger, F. A 1-60 GHz 9.6 mW 0.18 V Output-Swing Static Clock Divider Circuit in 45-Nm SOI CMOS. In Proceedings of the 2018 25th IEEE International Conference on Electronics, Circuits and Systems (ICECS), Bordeaux, France, 9–12 December 2018; pp. 41–44. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.