

Scalable Resonant Power Clock Generation for Adiabatic Logic Design

Ragh Kuttappa, Leo Filippini, and Baris Taskin
ECE Department, Drexel University, Philadelphia, PA, USA

Abstract—In this paper, a scalable and passive component less power-clock generation for adiabatic logic circuits inclusive of the adiabatic core is presented. The power-clock is traditionally a sinusoidal signal, that acts as the power and timing source to adiabatic gates. The slope of the power-clock signal determines the overall energy efficiency of the adiabatic gates. Prior works have considered using LC based power-clock generation circuits, which are often considered costly in terms of high area and high-Q inductor requirements. In this paper, a scalable solution is presented that features a unique innovation for the power clock generation circuit in step-charged circuits designed with resonant clocking. Resonant rotary style clocking is used to provide low power control to step-charged power clock circuits. SPICE based simulations are performed at 400 MHz on large scale adiabatic implementations of LGSynth'91 benchmark circuits. The overall energy efficiency of the proposed implementations are compared to those of equivalent adiabatic circuits with a traditional LC power-clock solution.

I. INTRODUCTION

Adiabatic logic achieves low power consumption by recycling the energy dissipated back to the power source [1]. In adiabatic logic, the transition from one voltage level to another is not abrupt like standard CMOS logic. A voltage ramp (power-clock) is utilized to charge and recover the energy from the output. A sinusoidal signal, called the power-clock, is used as the voltage ramp which serves both as an energy and timing source to adiabatic gates. The power-clock enables charge to flow back and forth from the load, recycling part of the energy back to the power-clock resulting in low power consumption. Depending on the adiabatic logic family, there can be two or more power-clock signals, referred to as phases of the power-clock.

The power-clock eliminates the need for registers in a pipeline, because it provides timing to the adiabatic gates, referred to as inherent pipelining [1]. Due to inherent pipelining each gate has a fixed latency; multi-level logic optimization cannot be used without significantly increasing the latency of a circuit. To enable wide spread adoption of adiabatic logic, design automation methodologies in the context of logic synthesis is required, as explored in [2]. In this work, logic synthesis for a four-phase adiabatic logic family energy efficient charge recovery logic (ECRL) along with power-clock circuit design is presented. The operation of the ECRL gate is similar to that of a sense amplifier, with the two cross-coupled PMOS transistors that amplify the voltage difference between

the two pull-down networks (PDNs). The output either follows the power-clock PC or remains at ground, due to the PDNs that are built with NMOS transistors. The pull-up activity of the output nodes is due to the PMOS transistors, and this allows ECRL to accommodate larger fan-ins than it is possible with static CMOS, when using minimum size devices [1]. The charge recovery logic presented in [2] leverages the large fan-in property to enable competitive operating frequency in comparison to conventional CMOS implementations.

Prior works have only provided solutions focusing on either power-clock generation or an energy efficient logic family [3–7]. This work presents a completely adiabatic system that includes the logic synthesis for adiabatic core and a power-clock generation and distribution scheme. Rotary traveling wave oscillators (RTWO) are utilized to generate the control signals for the step-charged circuits that provide the power-clocks for the adiabatic gates [8, 9]. RTWOs are designed with CMOS interconnects and inverter pairs, and provide high frequency, multiple phase, and low-power clock signals. The RTWOs are distributed in an array topology across the design providing control signals over the entire die area of the benchmark circuit. The adiabatic system presented in this work is implemented on the LGSynth'91 combinational benchmark circuits. In simulation based studies, the power consumption, conversion efficiency, and overall area overhead results are presented. The overall energy efficiency and area overhead presented in this work is compared to an adiabatic system implemented with an inductor (L) and capacitor (C) based power-clock generator.

This rest of this paper is organized into the following sections. Technical background is discussed in Section II. The adiabatic system designed in this work is presented in Section III. Experimental analysis and the results are discussed in Section IV. Conclusions are provided in Section V.

II. PRELIMINARIES

A. Adiabatic Principles

In Fig. 1, two different ways of charging and discharging a capacitance C_L through a resistance R are illustrated. Abruptly charging a capacitance C_L illustrated in Fig. 1(a) results in an energy profile estimated as:

$$E_C = C_L V_{DD}^2, \quad (1)$$

where V_{DD} is the magnitude of the square wave. The adiabatic charging of the capacitance C_L with a trapezoidal signal

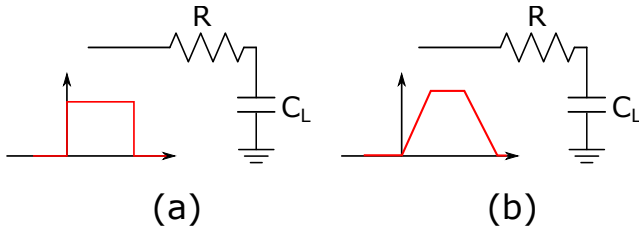


Fig. 1. Charging a capacitance. (a) Abruptly and (b) Adiabatically.

illustrated in Fig. 1(b) has an energy profile estimated as:

$$E_A = 8 \frac{C_L^2 V_{DD}^2 R}{T} \quad (2)$$

where V_{DD} and T are the peak-to-peak voltage and period of the trapezoidal-wave (power-clock), respectively. For $T \rightarrow \infty$, the energy dissipated on the resistance R ideally becomes *zero*. If a sine-wave is used in place of the trapezoidal-wave, Eq. (2) becomes:

$$E_A = \frac{\pi^2}{2} \frac{C_L^2 V_{DD}^2 R}{T} \quad (3)$$

where V_{DD} and T are the peak-to-peak voltage and period of the sine-wave (power-clock), respectively.

In order for ensure an adiabatic system consumes lower energy than other conventional systems, the ratio $E_A/E_C < 1$.

B. Adiabatic Logic Design

Charge recovery principles enable the design of different adiabatic logic families. Adiabatic logic families are categorized based on the number of power clocks required. The power-clock serves as the power and timing source for the adiabatic logic families. On the contrary, conventional CMOS logic employs flip flops to ensure timing is met across a path. The charge recovery principles at the gate level are largely known for a variety of logic families however, requires custom

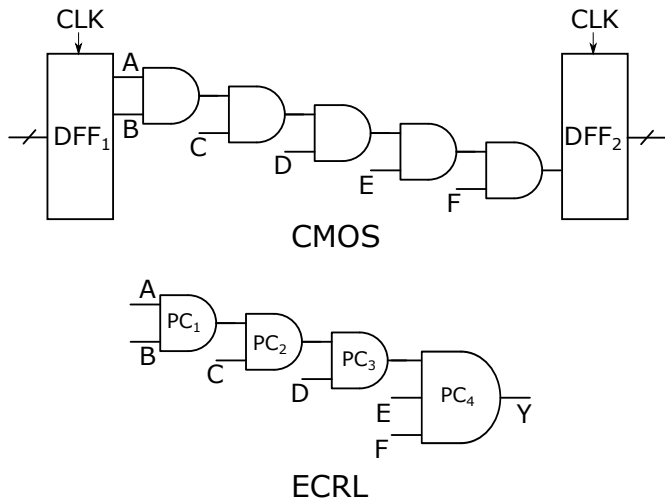


Fig. 2. Boolean function implementation for CMOS and ECRL logic.

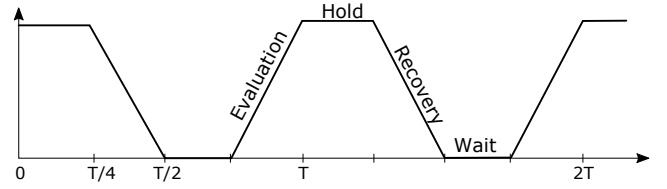


Fig. 3. Power clock illustrating the phases of operation for adiabatic logic.

methodologies for widespread use. ECRL at a higher level, in terms of latency and throughput of a pipeline requires dedicated synthesis methodologies. In Fig. 2, two equivalent pipelines, in conventional CMOS and ECRL with identical latency are illustrated. The data in the CMOS pipeline of Fig. 2(a) propagates from DFF₁ to DFF₂ in one CLK period. The data input from the left of Fig. 2(b) propagates to the right side through phases PC₁ to PC₄ in one power-clock period. In order to ensure the two circuits have the same end-to-end latency, the logic gates require dedicated logic synthesis for adiabatic gates, as explored in [2].

C. Power-Clock Generation

The power-clock is an oscillating signal the provides the supply voltage V_{DD} to the adiabatic logic. Depending on the adiabatic logic family there can be power-clocks ranging from $\phi_1 - \phi_4$. Each power clock has four operation zones: Evaluation, Hold, Recovery, and Wait, illustrated in Fig. 3. For adiabatic logic families that require more than one power-clock, each power clock is phase shifted by 90° to ensure all signals are constant during the evaluation phase. Prior works have focused on designing power-clock generation circuits with active and passive components [4, 6, 10]. Depending on the type of components utilized efficiency, scalability, and feasibility have been the major challenges.

1) *Quadrature Oscillator*: Inductor (L) and capacitor (C) based oscillators have been utilized to generate multiphase power-clocks for adiabatic logic circuits [11]. A quadrature oscillator is employed to generate four phases of the power-clock for ECRL gates. The resonant frequency of the LC-tank is estimated as:

$$f_n = \frac{1}{2\pi} \frac{1}{\sqrt{LC}}. \quad (4)$$

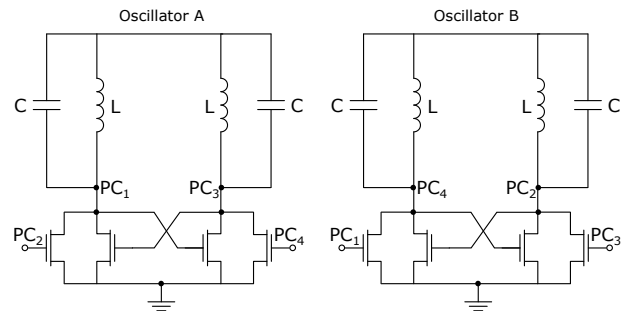


Fig. 4. LC-tank quadrature oscillator to generate 4 phases of power clock.

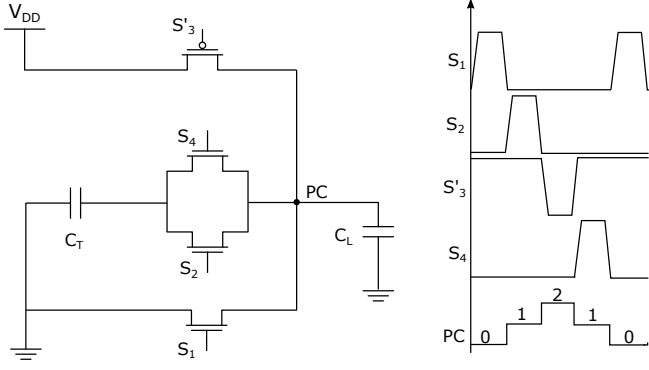


Fig. 5. Stepwise charging scheme for power clock generation.

The efficiency of the LC based oscillator is directly related to the load that is driven by each power-clock. Fluctuations in the load and varying data patterns decrease the efficiency of the LC oscillator [1]. In addition to the losses in the adiabatic circuit, deviations in the load capacitance leads to a deviation in the resonant frequency of the LC oscillator [1].

2) *Step charged based Power-Clock Generator*: An alternative solution to LC based oscillators is stepwise charged (SC) power-clock generators. In SC power-clock generators, the peak-to-peak voltage is charged in steps rather than abruptly. Varying implementations with different number of steps (N) have been researched [1, 10, 12]. In Fig. 5, a 2:1 SC power-clock generator is illustrated along with the control signals. The energy dissipation of a SC power-clock is estimated as:

$$E_{SC} = \frac{C_L V_{DD}^2}{N}. \quad (5)$$

It has been shown that increasing the number of steps results in less energy consumption [1]. However, the energy required to control the increase in number of switches negating any savings in energy. The overall losses due to controlling the switches with an increase in number of steps is estimated as [1]:

$$E_{SC} = \left(\sum_{i=1}^N C_i + \sum_{i=0}^N C_i \right) V_{DD}^2, \quad (6)$$

where C_i is the capacitance of the switch i . In addition to the increase in the number of switches, to generate 4-phases of the power-clock would require 4 SC circuits. The area overhead for the control logic and tank capacitors increases with the increase in number of power-clock phases required.

III. THIS WORK

In the following sections, design automation for charge recovery logic (CRL) synthesis and power-clock generation are presented.

A. Adiabatic Circuit Generation

Designing adiabatic logic gates with dedicated logic synthesis tools to ensure competitive latency matching with conventional CMOS gates was first studied in [2]. The conventional

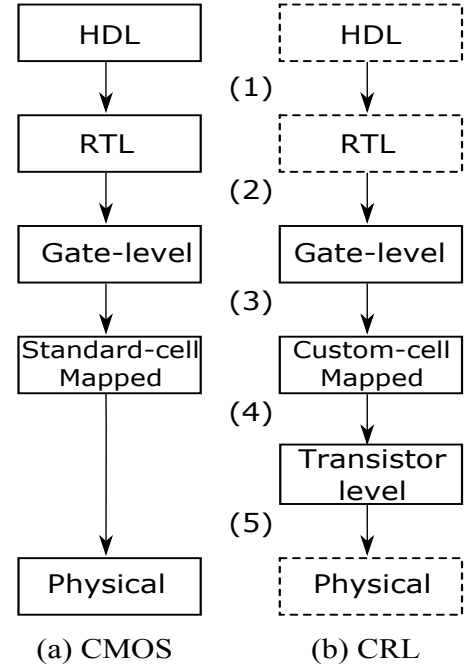


Fig. 6. A simplified synthesis flow used for CMOS, and the CRL synthesis flow. Dashed lines represents parts of the flow that are not yet implemented.

CMOS synthesis flow is illustrated in Fig. 6(a) showing the five steps:

- 1) **High-Level Synthesis**: Convert from a Hardware-Description-Language (HDL) to the next lower description: Register-Transfer-Level (RTL). A series of resource scheduling, logic minimization, logic optimization, and IP usage happens at this step.
- 2) **Synthesis and Optimization**: Translate the RTL description into a netlist of technology independent logic gates. A number of optimization algorithms can be used at this step to increase performance, reduce area, et cetera.
- 3) **Technology Mapping**: Convert the generic gate-level netlist into a technology-specific netlist, using standard cells. A number of technology mapping algorithms are used in order to obtain a mapped netlist that meets specifications (power, area, performance).
- 4) **Schematic Generation**: In the CMOS flow, standard cells have a layout that is already characterized, so there is no need to generate it.
- 5) **Layout Generation**: The layout of each mapped standard-cell is placed, logic gates are then connected to each other. Place & Route algorithms, floorplanning, Clock-Tree-Synthesis and other operations are carried out during this step.

CMOS synthesis flows have been actively researched and optimized for a number of years. To generate custom synthesis flows for CRL, primary focus has been on the following steps [2]:

- 3) **Custom-Cell Mapping**: a compression algorithm is used to cluster logic operations to be implemented in large fan-in

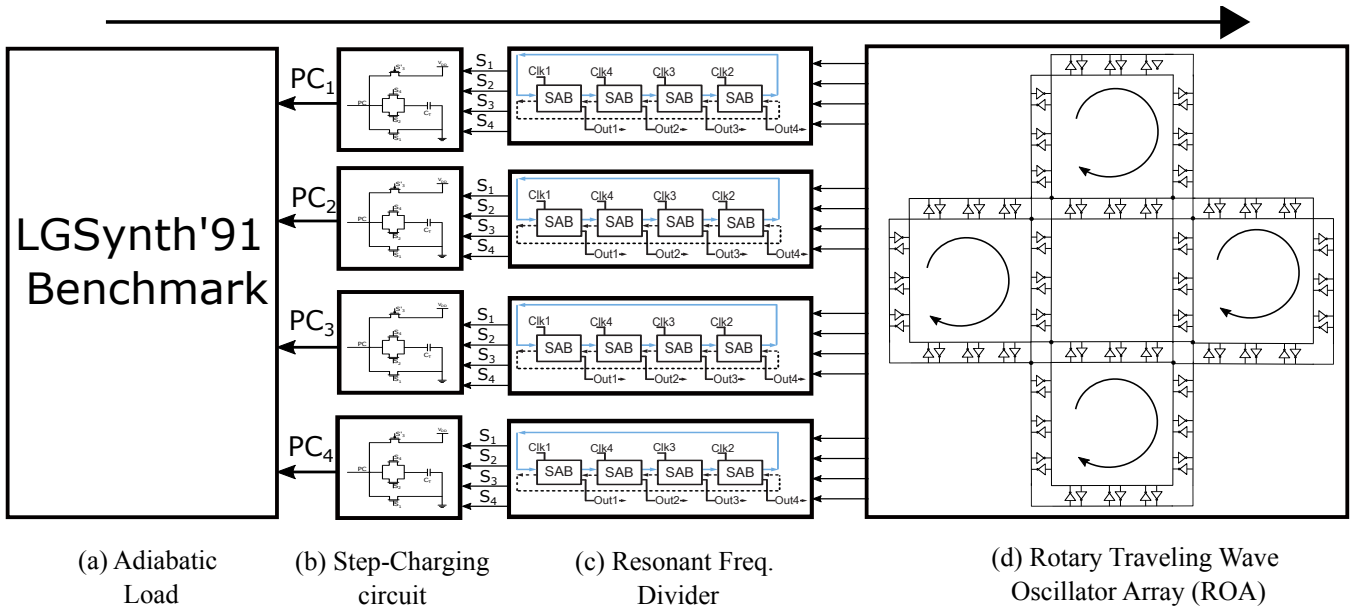


Fig. 7. Proposed multiphase clock generation and delivery architecture for adiabatic logic.

gates, taking care that data dependencies are met. The input to this algorithm is an and-inverter-graph (AIG). Since charge recovery logic cannot take advantage of multi-level logic minimization, this algorithm does not map logic functions to a set of standard cell, but rather creates custom cells to take advantage of the large fan-in capability of CRL.

4) Schematic Generation: a recursive algorithm is used to convert the boolean description of each gate from previous step to a spice netlist that can be simulated.

The CRL synthesis flow and the EDA tool are targeted for combinational logic synthesis [2]. CRL gates, have a fixed delay that only depends on the selected logic family, independent of the number of inputs: in ECRL, for example, each logic gate has a delay of $T/4$, where T is the period of the power-clock. Regardless of the technology node, the outputs of an ECRL gate will have set to a logic zero or one after a delay of $T/4$. However, the technology node and the number of inputs impacts the maximum frequency at which the power-clock can be operated before a logic gate will fail to represent the boolean function that it implements.

ECRL supports high fan-ins, larger boolean logic functions can fit into a complex gate in a CRL circuit in order to reduce the logic-depth, which in turns reduces the delay of the CRL circuit. Maintaining a low delay of a CRL circuit is useful because it allows i) existing designs to be easily converted from CMOS to CRL, and ii) recursive operations to be efficiently implemented. By leveraging the fact that CRL gates can have large fan-ins, the synthesis algorithm merges boolean functions into each other, reducing the logic depth of the circuit. In this manner, large logic gates are created and implemented with a custom boolean-to-netlist algorithm [2].

B. Resonant Rotary based Power-Clock Generator

Stepwise charged circuits are employed to generate the power-clocks for the ECRL gates in this work. A rotary traveling wave oscillator (RTWO) and resonant frequency divider are utilized to generate the control signals for the step-charged (SC) power-clock generator. The master clock frequency of the RTWOs remain unaltered after design as the control signals are generated from the resonant frequency dividers. The resonant frequency dividers that control the switches in the SC power-clock generators enable greater power savings due to them being adiabatic circuit blocks. Based on the number of switches required for the step-charged circuit the resonant frequency divider topology is designed (Fig. 7(c)). The rotary traveling wave oscillator is designed in an rotary oscillator array (ROA) with the frequency divider topology information and target frequency (Fig. 7(d)). The two-step charged circuit is designed as illustrated in Fig. 7(b) (detailed in Fig. 5). MOS type capacitors are used for the tank capacitors along with three NMOS switches and one PMOS switch.

The two-step charged circuit requires four control signals, illustrated in Fig. 5. In order to generate the control signals, a divide by three resonant frequency divider is designed, illustrated in Fig. 7(b). The divider topology requires four clocks with a constant phase delay of $1/4 \times \pi$ between them. The adjacent phase delays between the clocks are set to be $3/4 \times 2\pi$, resulting in a phase delay of $3/4 \times 2\pi$ between each SAB. As the clock signal loops through the frequency divider, the total phase delay is $4 \times 3/4 \times 2\pi$, generating a divide by three signal. To generate the 90° phase delay between the power clocks, the routing order between outputs of the step-charged circuit is modified.

Routing the generated power-clock to the ECRL gates is

TABLE I
LGSYNTH'91 COMBINATIONAL BENCHMARK CIRCUITS POWER
CONSUMPTION. $V_{DD} = 1$ V AND FREQUENCY = 400 MHz.

LGSynth'91 Benchmark Circuit	Norm. Power (%Eff.)		Norm. PC Area	
	SC Implem.	LC Osc. Implem.	SC + RTWO	LC Osc.
I9	1 (45%)	1.10 (45%)	1	1.3
C1355	1 (45%)	1.10 (45%)	1	1.3
Too_large	1 (47%)	1.20 (40%)	1	1.2
Alu4	1 (48%)	1.25 (40%)	1	1.4
Rot	1 (50%)	1.30 (40%)	1	1.4
X3	1 (50%)	1.35 (38%)	1	1.3
C1908	1 (40%)	1.35 (40%)	1	1.3
Frg2	1 (45%)	1.40 (40%)	1	1.3
Average	-	1.26	-	1.3

critical to ensure the phase is matched across the entire design. Steiner tree routing is employed to route the power-clock from each SC power-clock generator to the corresponding ECRL gate. Consider the tapping point for a particular phase Θ_{P_i} to be located at (x,y). The power-clock source taps onto Θ_{P_i} that satisfies the phase requirement. The SC power-clock generators are placed across the die to ensure: i) Θ_{PC_p} - the phase required for the ECRL gate and ii) Θ_{l_i} the phase attributed to the tapping wire l_i satisfy, $\Theta_{PC_p} = \Theta_{l_i} + \Theta_{P_i}$.

IV. EXPERIMENTAL ANALYSIS

The adiabatic system presented in this work is implemented on a subset of the LGSynth'91 benchmark circuit suite with the step-charged circuit as the power-clock generator in the 65 nm technology node. The control signals for the SC power-clock are generated from RTWOs implemented with the top 2 metal layers using HFSS. The tank capacitors for the SC power-clock are of MOSCAP type. An LC based quadrature oscillator [11] is implemented in the 65 nm technology node to compare the energy efficiency and power-clock area of the proposed system. The results are presented at an operating frequency of 400 MHz with a supply voltage of 1 V. All the circuits in the selected benchmark suite are simulated with random input patterns with an activity factor of 0.1, or 10 %, which is typically used in system-level analysis [13]. The power consumption of the overall system and the power-clock area is compared with quadrature LC oscillator based power-clock designs in Table I.

In Table I, the power consumption is reported along with the conversion efficiency of the adiabatic system. The efficiency of the quadrature LC oscillator operating at 400 MHz is 75%. The load of the adiabatic system acts as a capacitive load to the LC oscillator, decreasing the efficiency to 45% in the case of the I9 benchmark circuit. The efficiency of LC oscillator degrades due to the deviation from the resonant frequency of the LC tank owing to the switching activity of the adiabatic system and the additional load being driven. In addition, the overall efficiency of the system is presented in Table I. The SC based designs have higher conversion efficiency as the slope of the power-clock follows Eq. (2). On an average the LC oscillator based implementations consume 26% more power

when compared to the SC based implementations presented in this work.

In Table I, the total area of the power clock generation circuits are presented. The area reported for the power-clocks include the power clock generation circuit and control signal circuitry. On an average the LC based power-clock generator occupies 30% more area when compared to the SC based power-clock generator presented in this work.

V. CONCLUSION

In this paper, an adiabatic system with power clock generation is presented. The adiabatic system is implemented with a dedicated logic synthesis flow considering latency and throughput requirements. The LC oscillator based implementation of the adiabatic system presented in this work consumes 26% more power when compared to the SC based implementation leveraging RTWOs to control the switches of the SC power-clock generator. In addition, the LC oscillator based implementation occupies 30% more area when compared to the SC based implementation.

REFERENCES

- [1] P. Tiechmann, *Adiabatic Logic - Future trend and system level perspective*. New York, NY: Springer, 2012.
- [2] L. Filippini, "Charge recovery circuits," *Thesis, Drexel University*, 2019.
- [3] M. Arsalan and M. Shams, "Charge-recovery power clock generators for adiabatic logic circuits," in *Proceedings of the International Conference on VLSI Design (VLSID)*, Jan 2005, pp. 171–174.
- [4] H. Mahmoodi-Meimand and A. Afzali-Kusha, "Efficient power clock generation for adiabatic logic," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 4, May 2001, pp. 642–645 vol. 4.
- [5] A. Bargagli-Stoffi, G. Iannaccone, S. Di Pascoli, E. Amirante, and D. Schmitt-Landsiedel, "Four-phase power clock generator for adiabatic logic circuits," *Electronics Letters*, vol. 38, no. 14, pp. 689–690, July 2002.
- [6] C. H. Ziesler, S. Kim, and M. C. Papaefthymiou, "Resonant clock generator for single-phase adiabatic systems," in *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED)*, Aug 2001, pp. 159–164.
- [7] W. C. Athas, L. J. Svensson, and N. Tzartzanis, "A resonant signal driver for two-phase, almost-non-overlapping clocks," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 4, May 1996, pp. 129–132 vol.4.
- [8] J. Wood, T. C. Edwards, and S. Lipa, "Rotary traveling-wave oscillator arrays: a new clock technology," *IEEE Journal of Solid-State Circuits (JSSC)*, vol. 36, no. 11, pp. 1654–1665, Nov. 2001.
- [9] R. Kuttappa, S. Khoa, L. Filippini, V. Pano, and B. Taskin, "Comprehensive low power adiabatic circuit design with resonant power clocking," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS)*, 2020, pp. 1–5.
- [10] H. S. Raghav, V. A. Bartlett, and I. Kale, "Energy efficiency of 2-step charging power-clock for adiabatic logic," in *International Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS)*, Sep. 2016, pp. 176–182.
- [11] A. Rofougaran, J. Rael, M. Rofougaran, and A. Abidi, "A 900 mhz cmos lc-oscillator with quadrature outputs," in *IEEE International Solid-State Circuits Conference (ISSCC)*, 1996, pp. 392–393.
- [12] L. J. Svensson and J. G. Koller, "Driving a capacitive load without dissipating fCV^2 ," in *Proceedings of the IEEE Symposium on Low Power Electronics*, Oct 1994, pp. 100–101.
- [13] N. H. E. Weste and D. M. Harris, *CMOS VLSI Design - A Circuit and System Perspective*. Pearson, 1996.