Modelling, Simulation and Verification of 4-phase Adiabatic Logic Design: A VHDL-Based Approach

Maheshwari, S., Bartlett, V. and Kale, I.

Accepted Manuscript

Modelling, Simulation and Verification of 4-phase Adiabatic Logic Design: A VHDL-Based Approach

Sachin Maheshwari, V.A. Bartlett, Izzet Kale

PII:	S0167-9260(18)30433-4
DOI:	10.1016/j.vlsi.2019.01.007
Reference:	VLSI 1553
To appear in:	*Integration*
Received Date:	14 August 2018
Accepted Date:	22 January 2019

Please cite this article as: Sachin Maheshwari, V.A. Bartlett, Izzet Kale, Modelling, Simulation and Verification of 4-phase Adiabatic Logic Design: A VHDL-Based Approach, *Integration* (2019), doi: 10.1016/j.vlsi.2019.01.007

Modelling, Simulation and Verification of 4-phase Adiabatic Logic Design: A VHDL-Based Approach

Sachin Maheshwari, V.A.Bartlett and Izzet Kale
Applied DSP and VLSI Research Group, Department of Engineering,
University of Westminster,
London, W1W 6UW, United Kingdom
Email: w1412503@my.westminster.ac.uk, {v.bartlett, kalei}@westminster.ac.uk

Abstract

The design and functional verification of the 4-phase adiabatic logic implementation take longer due to the complexity of synchronizing the power-clock phases. Additionally, as the adiabatic system scales, the amount of time in debugging errors increases, thus, increasing the overall design and verification time. This paper proposes a VHDL-based modelling approach for speeding up the design and verification time of the 4-phase adiabatic logic systems. The proposed approach can detect the functional errors, allowing the designer to correct them at an early design stage, leading to substantial reduction of the design and debugging time. The originality of this approach lies in the realization of the trapezoidal power-clock using function declaration for the four periods namely; Evaluation (E), Hold (H), Recovery (R) and Idle (I) exclusively. The four periods are defined in a VHDL package followed by a library design which contains the behavioral VHDL model of adiabatic NOT/BUF logic gate. Finally, this library is used to model and verify the structural VHDL representations of the 4-phase 2-bit ring-counter and 3-bit up-down counter, as design examples to demonstrate the practicality of the proposed approach.

Keywords— adiabatic logic; complexity; modelling; verification; VHDL

1. Introduction

The increased usage of battery-less applications (e.g. a smart card) and the rising energy density due to the technology shrinkage, energy-efficiency has become a major concern in the design of large and complex systems. A circuit technique, known as "Adiabatic Logic" based on the CMOS technology, has the potential for low energy operation albeit at some cost in terms of performance speed [1]-[3], [10]-[13], [17]-[30]. Thus, making it a worthy choice for smart card application. Several research papers exist that demonstrates the energy saving potential of the adiabatic logic technique in comparison to the non-adiabatic logic (conventional CMOS). In [1], the authors have shown that the sense-amplifier based adiabatic logic performs better than the non-adiabatic logic using 65nm, 45nm and 32nm Berkeley Predictive Technology Models (BPTM). Likewise, in [2] the behaviour of adiabatic logic circuits is analysed in the weak inversion/subthreshold regime. Through extensive post-layout simulation, the authors demonstrate that adiabatic circuits in the subthreshold region can save significant energy in comparison to its equivalent non-adiabatic implementation. Moreover, the recent work showing the applicability of the adiabatic principle on capacitive logic demonstrates the effectiveness of the technique to achieve zero-power dissipation in an ideal setting [3], [4]. Additionally, in [5], the authors presented a detailed analysis of CMOS and NEMS adiabatic logic performance including the power-clock generator. The analysis presented shows that the NEMS-based adiabatic logic is exceptionally suitable for low power operation, however, its performance is highly dependent on the switch series resistance. Nonetheless, the only feasible method for the functional verification of the adiabatic design at transistors level is by using SPICE simulation, which is inherently time-consuming and difficult. The design of the non-trivial adiabatic circuits demands much more effort in contrast to the non-adiabatic logic for which well-developed tools exist. The main distinction between the two lies in the fact that the former uses a slowly changing power-clock/supply whereas the latter works with DC (constant) power-supply.

1.1. Motivation

VHDL is efficiently used for the signal levels '0' and '1' having zero rise and fall times for ideal simulations. However, the waveforms are more complex in the adiabatic circuit design, due to the requirement of the trapezoidal 4-phase power-clock and the dual-rail encoding of inputs and outputs. Moreover, to realise a trapezoidal power-clock in VHDL, two more logic levels depicting the ramp are required to be modelled in addition to the logic '1' and logic '0' such that all the four periods share the same time length. Fig.1 shows the trapezoidal power-clock with four equal periods.

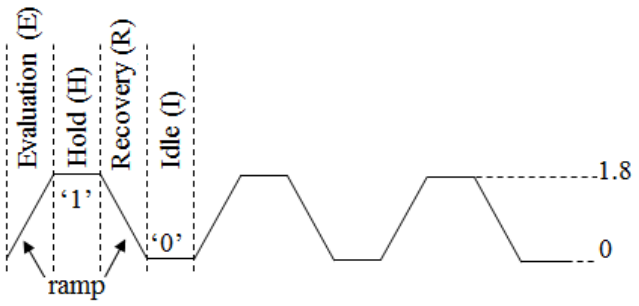


Fig. 1. Trapezoidal power-clock.

To the author's best knowledge and the literature review carried out, the first modelling in VHDL of adiabatic logic was done by M. Vollmer and J. Gotze in 2005. The authors described the adiabatic logic in VHDL for a systolic array with precise timing and bit-true calculation [6]. The work presented in the paper included the description of the logic blocks that requires 4-phase clocking scheme but, however, they used one global clock net in place of the 4-phase power-clock for cascade

logic designs. In addition, the authors also did not model the dual-rail encoding of the input and output signals. After a year, Laszlo Varga et.al, demonstrated a two-level pipelining scheduling of the adiabatic logic design using integer linear programming formulation and a heuristic scheduling [7]. In this work, the authors presented the VHDL description for the functional simulation of the synthesized adiabatic datapath along with the non-adiabatic portion of the digital system. This method mainly concentrated on producing a pipeline schedule of the power-clock behaviour of the adiabatic logic but did not model the power-clock and the dual-rail behaviour of the adiabatic logic. In 2010, David John Willingham [8] specified Asynchrobatic Logic in Verilog, an industry standard Hardware Description Language (HDL). The author first demonstrated the idea in a single-rail scheme and then extended it to a dual-rail. The latter was found to be absent in Vollmer and Laszlo's modelling. The modelling of the dual-rail inputs and outputs found to be beneficial in detecting an invalid circuit operation to some extent. Three states namely; valid (logic '1' or logic '0'), invalid and inactive were defined. This approach too did not model the trapezoidal power-clock in HDL, rather, the 4-phases of the power-clock were generated using the square waves. Although all the above-mentioned papers demonstrated the pipeline timing, none observed the adiabatic principles, i.e., the circuit produces a valid output signal when the input and the power-clock supply signal are at 90^o phase difference. Additionally, the power-clock is depicted by a square-wave which combines the evaluation and hold period of the trapezoidal power-clock as logic '1', whereas, the recovery and idle periods as logic '0'. Therefore, the use of a square wave will cause the circuit to malfunction if a significant clock skew and/or delay between the power-clock and the input logic signal exists.

1.2. Contribution of this paper

The focus of this paper is to develop a VHDL-based modelling approach for the functional simulation of the 4-phase adiabatic logic circuits such that the design and validation time is reduced in a large complex adiabatic system. The main contributions of this paper are as follow;

1) Shortcomings of the existing HDL-based approach is discussed and a new approach using VHDL is presented.
2) The exact behaviour of the trapezoidal power-clock is represented by modelling all the four periods distinctively using VHDL.
3) The proposed approach detects the circuit's invalid input operations by monitoring the generated complementary outputs.
4) The proposed approach includes the modelling of the dual-rail input and output signals.
5) The functional aspects of the proposed approach are verified for a 2-bit ring counter and a 3-bit up-down counter.

1.3. Structure of the paper

Section 2 describes the 4-phase adiabatic logic technique. Section 3 presents the existing and the proposed HDL approach for modelling adiabatic logic. The digital simulation using the proposed approach for adiabatic logic is presented in section 4. This section demonstrates the method of encoding trapezoidal waveforms using VHDL, conversion of dual-rail pulse-input to adiabatic signals and encoding of invalid inputs for precise timing. A comparison between the SPICE simulations of the PFAL NOT/BUF gate with that of the proposed modelling approach is also presented in this section. Section 5 demonstrate the comparison and validation of the proposed VHDL modelling with SPICE simulation results based on the case study of a 2-bit ring-counter and a 3-bit up-down counter. The conclusion of this article is presented in section 6.

2. Adiabatic Logic Technique

For devices such as smart cards, working around 13.56 MHz frequency, the adiabatic logic technique seems to be one of the attractive solutions to achieve low-power operation [9]. The energy dissipation of the adiabatic circuits infinitesimally tends to zero if the switching time is slowed down. It achieves reduce energy dissipation by the use of a slowly changing power-clock which allows approximately constant current charging/discharging thus, avoiding current surges. A decreased energy dissipation with increased switching time is, therefore, the defining property of an adiabatic switching [10]-[13]. In addition, adiabatic logic technique gives an additional advantage of recovering the stored energy back to the power-clock during the discharging/recovery period of the power-clock by the use of a power-clock generator [14]-[20]. The power-clock generator can be implemented either using a stepwise charging circuit [14]-[16] or using resonant inductor-based circuit [17], [18].

The mathematical relationship for the energy dissipation also known as Adiabatic Loss (AL), E_D, using a ramp during the charging phase is given as;

$$E_D = \frac{R_{ON}C_L}{T_r} C_L V_{DD}^2 \qquad (1)$$

Where C_L is the lumped load capacitance at the output node of the circuit, R_{ON} is the resistance of the charging path and V_{DD} is the maximum supply voltage. The detailed derivation of (1) can be found in [12]. According to (1), it is possible to reduce the energy dissipation to an arbitrary degree by increasing the ramping time to ever-larger values. However, there is a practical lower limit to the ramping time value due to the increased leakage at longer ramping times.

For more than two decades, adiabatic logic has been widely studied and various adiabatic logic families have been proposed [21]-[26]. Single phase [21] and 2-phase [22] adiabatic logic families are less complex due to the low complexity of power-clock generator but, have high computation time [23]. In 1996, A. Vetuli et al. [24] presented Positive Feedback Adiabatic Logic (PFAL), as a new adiabatic logic family, which makes use of a CMOS positive feedback amplifier. It is very similar to IECRL [25], but its evaluation tree is connected between the power-clock and the outputs as shown in Fig. 2.

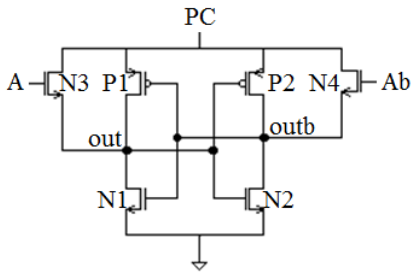


Fig. 2. PFAL buffer [24].

PFAL has been chosen for the comparison with the proposed modelling approach. In adiabatic logic, logic '1' is represented

by the waveform shown in Fig.1 whereas logic '0' represents the signal at the ground potential. In Fig. 2, when input 'A' is logic '1' its complementary input signal 'Ab' will be at the ground potential. The transistor N3 turns ON when signal 'A' crosses the threshold voltage of N3 and the output node 'out' start following the power-clock. When the output node 'out' reaches the threshold voltage of the transistor N2, it is turned ON and the node 'outb' is held at the ground potential. When PC is ramping down, during the recovery period, the input 'A' has already settled to ground. The node 'out' follows PC until it reaches the threshold voltage of the transistor P1. The residual charge on the 'out' node will be discharged non-adiabatically in the next power-clock cycle.

Other 4-phase adiabatic logic families such as IECRL [25], EACRL [26] and ECRL [27] is also applicable for the comparison. These adiabatic logic families also suffer from Non-Adiabatic Loss (NAL) arising because of the threshold voltage degradation. This part of the non-adiabatic discharge is independent of the frequency but, however, it can cause high energy dissipation for large adiabatic system designs with high fan-out and is defined by (2) as E_{NAL};

$$E_{NAL} = \frac{1}{2} C_L V_t^2 \qquad (2)$$

According to the adiabatic principle, it is important that the inputs remain at the same logic state during the rising or falling of the power-clock supply to avoid any non-adiabatic energy dissipation. Similarly, with 4-phase power-clocking scheme input and the power-clock signals must have 90° of phase difference with each other which is shown in Fig. 3. The power-clock can be broken down into four equal time periods namely; Evaluation (E), Hold (H), Recovery (R) and Idle (I). All the nodes in an adiabatic logic should follow the same adiabatic principle of charging and discharging in order to achieve low energy dissipation. For instance, in the cascade logic during the evaluation period of the power-clock (PC1), the input must be at the same logic state to produce a valid output in the hold period. Similarly, the second stage will sample its input while the output of the first stage is at the same logic state (hold period) and so on. This behaviour forms a pipeline, such that at every power-clock phase one input and one output is processed.

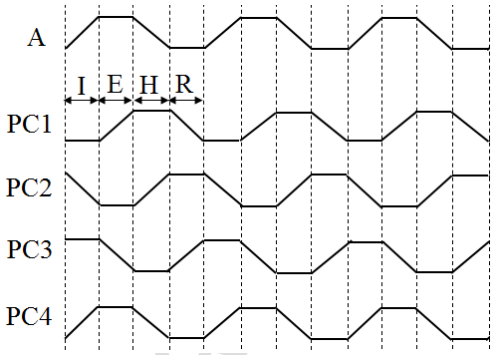


Fig. 3. 4-phase power-clocking scheme and input signal.

Recently proposed work on the performance comparison of the adiabatic logic families working in single-phase, 2-phase and 4-phase power-clocking scheme [28], [30] and on the power-clock generator [14], [15] have contributed to the knowledge base and dispelled a myth on the adiabatic logic capability of providing an energy efficient alternative to the non-adiabatic logic. In [28], [30] the authors show that the 4-phase adiabatic system is the most promising in terms of performance and energy requirement in comparison to the single and 2-phase adiabatic logic designs. The 4-phase adiabatic logic is implicitly pipelined and glitch-free. This suggests that the critical path is not a concern. Though various 4-phase energy-efficient adiabatic logic families have been proposed in the past two decades, where each encompasses many novel ideas and saves considerable energy compared to the non-adiabatic logic, the literature still lacks addressing the issues related to fast design and validation of the adiabatic logic used in a complex system design as accurately as possible

3. Encoding of HDL models

The most difficult part of modelling the adiabatic logic using conventional HDLs is that these languages are made entirely for encoding two logic levels ('0' and '1') and is either 'level' or 'edge' sensitive. With the adiabatic logic, having trapezoidal power-supply, the gates operating during a specific period must follow the adiabatic principle. In this work, we categorized two encoding styles: previously used voltage-level event-based encoding style and the proposed multi-level event-based encoding style.

3.1 Voltage-level Event-based Approach

So far voltage-level event-based approach is commonly used for encoding the behaviour of the power-clock in adiabatic logic which is similar to the non-adiabatic logic designs. It uses two logic levels to represent the four periods of the adiabatic power-clock. The logic '1' of the power-clock signal represents the evaluation (E) and hold (H) period of the trapezoidal power-clock, whereas, the logic '0' represents the recovery (R) and idle (I) period of the trapezoidal waveform. Fig. 4 shows the voltage-level encoding style. The adiabatic logic modelled using this style uses the clock signal as the power-clock.

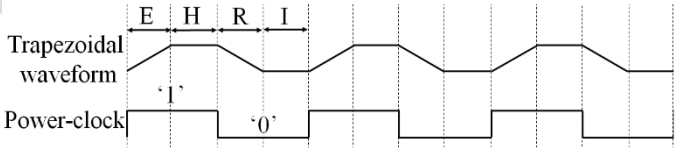


Fig. 4. Voltage-level event-based encoding.

3.2 Multi-level Event-based Approach

In order to improve the existing HDL models, a multi-level event-based approach is proposed to model the four equal time-periods of a trapezoidal power-clock. In this approach, the hold and idle periods of the power-clock are denoted as logic '1' and logic '0' respectively. On the other hand, the evaluation and the recovery period are encoded with an intermediate state marked as 'X'. It should be noted that both the evaluation and the recovery periods share the same duration of the ramp. This approach is not straightforward, as apart from generating the power-clock with three logic levels, the adiabatic inputs must also be generated with three logic levels for correct functionality and timing analysis. Fig. 5 shows the trapezoidal power-clock encoded using three logic levels.

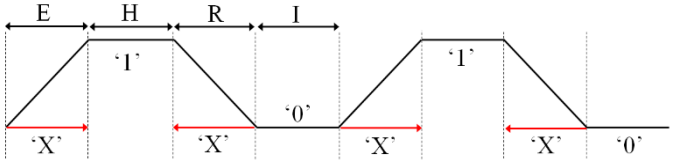


Fig. 5. Multi-level event-based encoding.

4. VHDL-based Approach for 4-phase Adiabatic Logic

One of the advantages of HDL modelling is that the design can be simulated with logic simulators and can be interfaced and mixed with the non-adiabatic logic designs for effective design solutions. Apart from modelling the power-clock, the adiabatic inputs must also be modelled for proper interfacing and correct functionality. The trapezoidal power-clock used at the transistor level is first encoded in standard logic to capture the timing behaviour of the adiabatic logic. This is followed by interconnection modelling (pulse input to adiabatic conversion) and gate-level modelling.

4.1 Modelling Trapezoidal power-clock

To realize the trapezoidal power-clock in standard logic, a multi-level approach is proposed as depicted in Fig. 5. Four states are required for representing four-periods. Each state is encoded based on the logic levels required. The four states can be easily generated using two D flip-flops. For simplicity, we forced the flip-flop outputs externally using the clock signal 'CLK' as a two-bit counter generating four states. Fig. 6 shows the VHDL simulation for generating the adiabatic power-clock signal. This approach can also be easily applied for other power-clocking schemes such as single-phase, 2-phase and Bennett clocking used in adiabatic logic designs. However, it will not be straightforward to deploy in 2-phase and Bennett clocking due to the long idle period for the former [23] and long hold and idle period for the later [31] power clocking scheme. A part of the VHDL description of the power-clock generation is shown in Fig. 7. The 4-phases of the power-clock coded similarly with each successive Power-Clocks (PC) are assigned with a value shifted by one count.

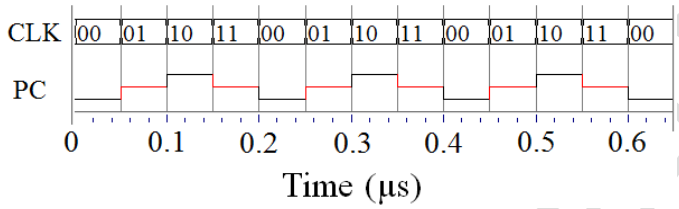


Fig. 6. VHDL simulation for generating a power-clock signal.

```
Process (CLK) is
 Begin
  if CLK ="00" then      //Idle Period//
       PC<= '0';
  elsif CLK = "01" then  //Evaluate Period//
       PC<= 'X'
  elsif CLK = "10" then  //Hold Period//
       PC<= '1';
  elsif CLK = "11" then  //Recovery Period//
       PC<= 'X';
  End if;
End Process;
```

Fig. 7. A part of the VHDL code for a power-clock generation.

The four periods of the power-clock are defined in a package as a function called "Adiabatic Signal". The package is shared between different VHDL models to develop the cell library of the basic adiabatic logic gates. A function defining the EVALUATE_edge is shown in Fig. 8.

```
Function    EVALUATE_edge    (Signal    s    :
 std_ulogic) Return Boolean is
  Begin
     Return  (s'event  AND  (To_X01(s)  =  'X')
     AND (To_X01(s'last_value)= '0'));
End Function;
```

Fig. 8. User-defined power-clock period as a function.

4.2 Conversion of the dual-rail pulse input to adiabatic inputs

One of the key requirements for the adiabatic logic to perform correctly is the generation of adiabatic inputs using a multi-level encoding approach. In an adiabatic logic, the input must be in the same logic state (hold period) during the evaluation period of the power-clock. This behaviour is captured in the proposed modelling by having the inputs to arrive one phase earlier than the power-clock such that a more realistic modelling, representing the adiabatic logic is represented. Similar to the modelling of the power-clock, the pulse input to adiabatic conversion also requires four states and depending on the input pulse (IN, INb) having logic levels '0' or '1', the complementary adiabatic signals (A, Ab) are generated. An equivalent dual-rail adiabatic outputs (A, Ab) are generated from the dual-rail pulse signals (IN, INb). Fig. 9 shows the VHDL simulation for generating adiabatic input signals for one power-clock phase. Similarly, the adiabatic inputs are generated for other power-clock phases.

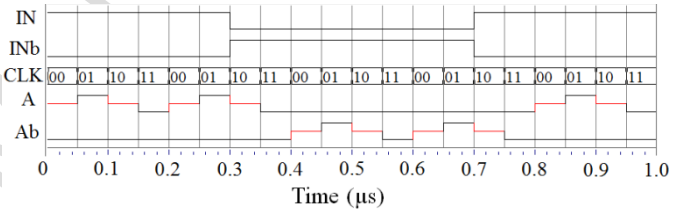


Fig. 9. Conversion of the dual-rail pulse input to adiabatic inputs.

4.3. Gate-Level Modelling

The VHDL modelling for an adiabatic NOT/BUF gate is collectively done using the power-clock generator, pulse input to adiabatic input (multi-level) conversion and the package defining the four periods of the power-clock. Using the proposed modelling, the conceptual block diagram of the adiabatic NOT/BUF gate is shown simplified in Fig. 10.

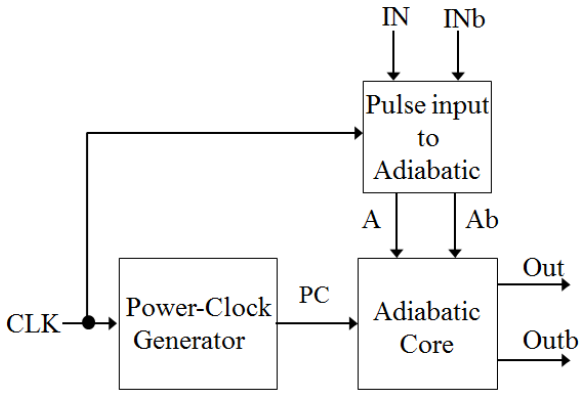


Fig. 10. A conceptual block diagram of an adiabatic NOT/BUF gate.

The timing behaviour of the SPICE simulation of the adiabatic logic was captured and is described in the flowchart, shown in Fig. 11. The flowchart represents the behaviour of how the adiabatic gate is modelled. First, the power-clock and the adiabatic inputs are generated from the pulse-input and then checked for the timing violation, such that it follows the adiabatic principle. Thereafter, the package containing the "Adiabatic Signal" is generated. All the above are then supplied to the adiabatic behavioural core which generates the dual-rail output.

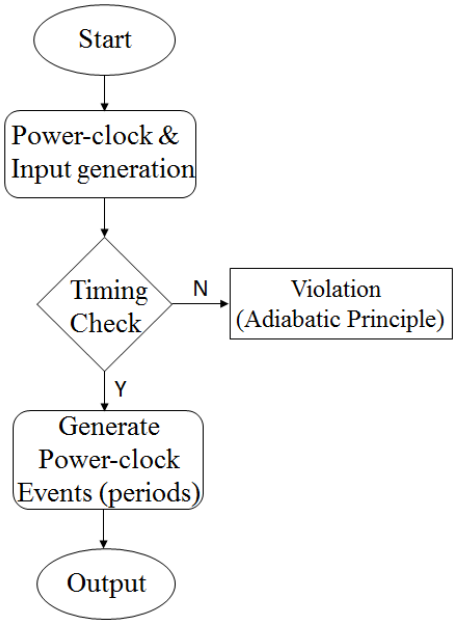


Fig. 11. Flowchart of the Adiabatic NOT/BUF gate.

The modelling of the adiabatic core is conceptualized for a dual-rail PFAL NOT/BUF gate generating the complementary outputs (Out, Outb). A fragment of the VHDL description of the NOT/BUF adiabatic gate is shown in Fig. 12. The code checks the invalid input condition in each of the four periods for any violation. The four periods of the power-clock are defined as the level sensitive signals.

The waveform of the adiabatic NOT/BUF gate encoded using VHDL is shown in the Fig. 13 (a). Similar to the transistor level design, the output follows the power-clock based on the input being processed. The proposed method produces the same behaviour as that of the SPICE level simulation. The four power-clock periods are defined in a package which is used in all the adiabatic VHDL description files by placing the 'USE' directive in the VHDL program. The output waveforms using the proposed VHDL modelling and the SPICE simulation are shown in Fig. 13 (a) and (b) respectively. The VHDL simulation shows the precise timing as depicted in the SPICE simulation. Moreover, the proposed modelling has an additional advantage of detecting an invalid input, which the SPICE simulations and the previously defined modelling approaches in the open literature fail to identify. The more detailed analysis of modelling an invalid state accurately is given in the next subsection.

For all the other adiabatic logic gates such as AND/NAND, OR/NOR, XOR/XNOR and MUX/DeMUX, it was assumed that the functional behaviour can be described similar to the adiabatic NOT/BUF gate. However, due to the dual-rail inputs in the adiabatic logic, the state-checking complexity for each power-clock period in the HDL behavioural code doubles for every increase in the number of input. As a result, the functional simulation and timing verification of all the gates with two inputs and above is performed by combining the functional part and the adiabatic NOT/BUF gate. The NOT/BUF gate used here helped in following the adiabatic principle and identifying the invalid complementary inputs while synchronizing the outputs for correct timing characterization.

```
Process (PC, A, Ab) is
 Begin
                // Idle Period
  if PC='0'  then
       Out <= PC;
       Outb<= PC;
  elsif  PC='0'  and  EVALUATE_edge  (A)  and
  EVALUATE_edge(Ab) then    // invalid state
       Out<= 'Z';
       Outb<='Z';
                // Evaluation Period
  elsif PC='X' and HOLD_edge (A) and HOLD_edge
  Ab) then                  // invalid state
       Out<= 'Z';
       Outb<='Z';
  elsif PC='X' and HOLD_edge (A) then
        Out <= PC;
        Outb<='0';
  elsif PC='X' and HOLD_edge (Ab) then
        Out <= '0';
        Outb<=PC;
                // Hold Period
  elsif  PC='1'and  RECOVERY_edge  (A)  and
  RECOVERY_edge (Ab) then   // invalid state
       Out <= 'Z';
       Outb<='Z';
  elsif PC='1' and RECOVERY_edge (A) then
       Out <= PC;
       Outb<='0';
  elsif  PC='1'  and  RECOVERY_edge  (Ab)  then
       Out <= '0';
       Outb<=PC;
                // Recovery Period
  elsif PC='X' and IDLE_edge (A) and IDLE_edge
  (Ab) then                 // invalid state
       Out <= 'Z';
       Outb<= 'Z';
  elsif PC='X' and IDLE_edge (A) then
       Out <= PC;
       Outb<='0';
  elsif PC='X' and IDLE_edge (Ab) then
       Out <= '0';
       Outb<=PC;
  End if;
 End Process;
End Behavioral;
```

Fig. 12. VHDL description of the NOT/BUF adiabatic gate.

Fig. 14 shows, for a two input AND/NAND gate, how the dual-rail VHDL is conceptualized as a logic function combined with a NOT/BUF gate for timing characterization. The collection of all the logic gates described in VHDL formed the cell library.

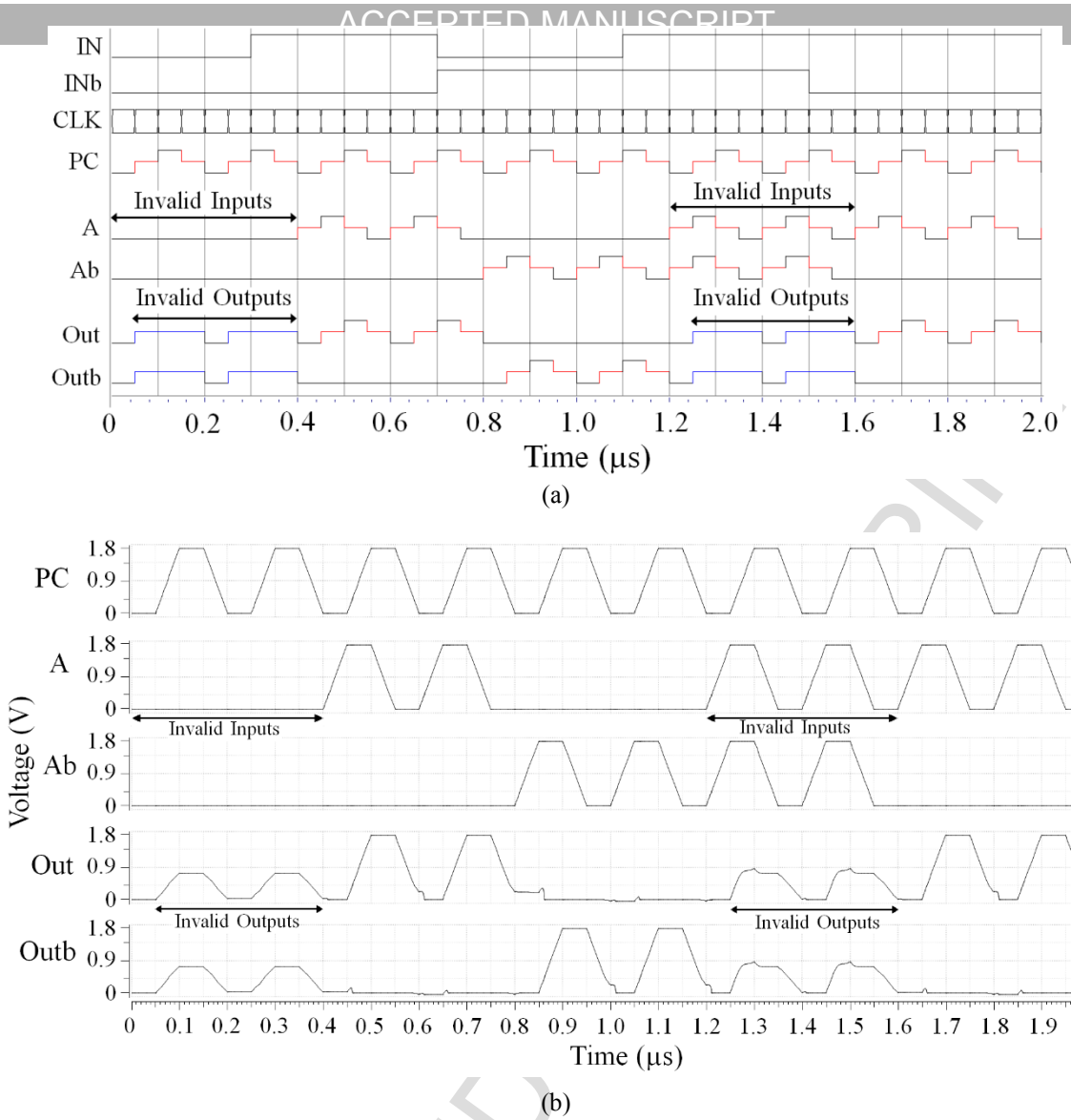


Fig. 13. Waveform results for PFAL NOT/BUF gate (a) VHDL Model (b) SPICE.

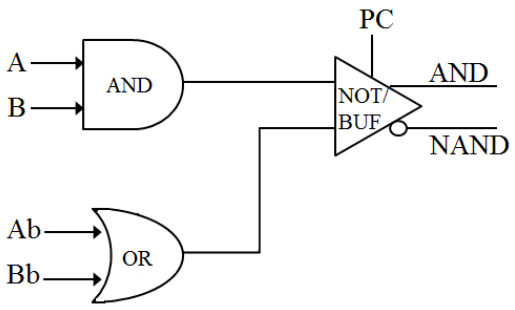


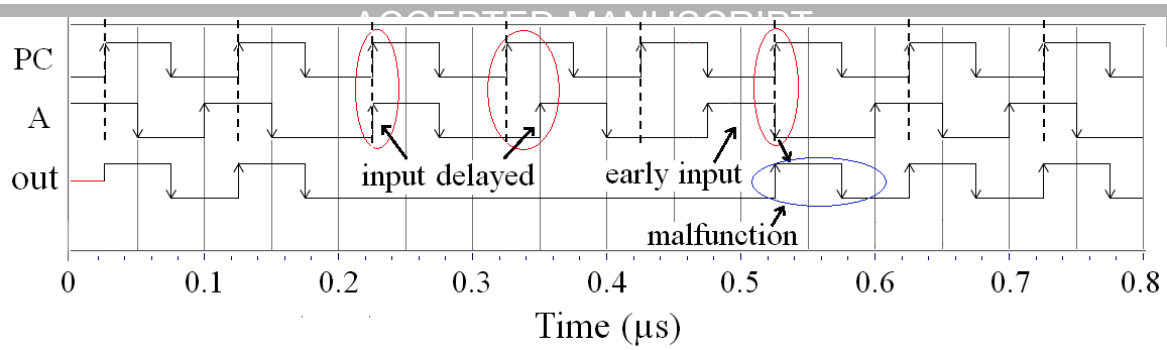
Fig. 14. Conceptualization of 2-input dual-rail AND/NAND gate for timing characterization.

4.4. Time Mapping

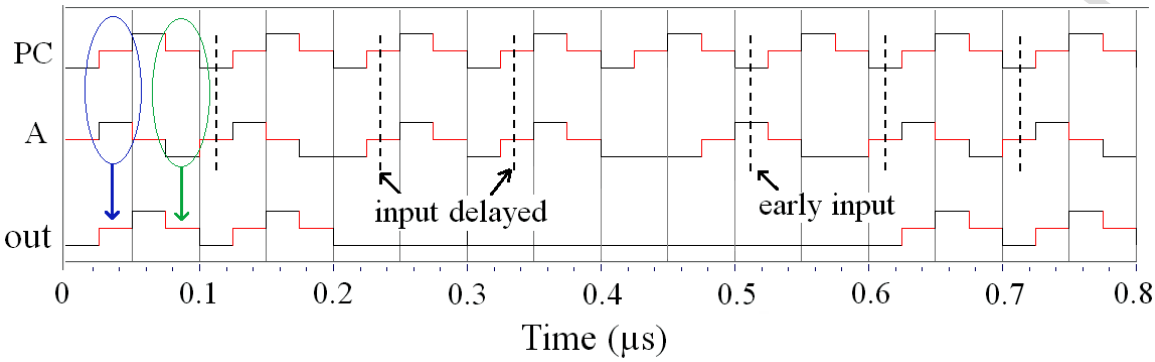
In the existing HDL modelling approaches, timing issues exist and circuit malfunction due to the clock skew and signal delays. This can be visualized in Fig. 15 (a), where it shows the waveforms of a NOT/BUF PFAL gate using square wave. For simplicity only, the signals which are logic '1' are shown. The power-clock is represented by PC, input 'A' is logic '1' whereas the signal 'Ab' is logic '0' (not shown). The output of the gate is 'out'. From Fig. 15 (a), it can be seen that an incorrect output occurs when the input arrives early with respect to the power-clock by a certain period. The same condition can occur if the power-clock is either delayed or arrives early. In a small circuit such errors can be detected easily, but for a large circuit, such errors will be time-consuming. To show the validity of our proposed approach same simulation was performed under similar conditions. From Fig. 15 (b) it can be seen that the proposed approach is robust as it produces a valid output when the adiabatic principle is violated by early and delayed input.

The operation of an adiabatic gate can be somewhat complex to model accurately. Although the trapezoidal power-clock has been modelled as precise as possible for all the four periods distinctly, the circuit can still malfunction if the invalid inputs and the power-clock synchronization issues [32] are not considered. Thus, it is essential to consider the invalid conditions and model them as accurately as possible for precise timing in VHDL.

The invalid conditions occur mainly due to the two cross-coupled inverters forming a latch (Fig. 2), which retains the last value stored on the complementary output nodes. For example: if the complementary inputs (A, Ab) are at logic '0' (invalid state), the outputs will retain the last value stored on it. This



(a)



(b)

Fig. 15. Example of simulated waveforms of delayed and early arrival of inputs. (a) The existing approach shows the violation for early input arrival (late power-clock arrival) and produces the wrong output (out), (adiabatic principle violation) (b) no violation using the proposed approach.

suggests that if the last value on the two output nodes (Out and Outb) is logic '1' and logic '0' respectively then the same value will be retained. The corresponding invalid condition is depicted in the Fig. 16. This, invalid input operation in a large circuit will be difficult to debug, specifically in the case when functionally, the logic value '1'and '0' is expected on the two output nodes. In addition, this invalid circuit operation will lead to high energy consumption, due to the non-adiabatic loss (NAL) [33]. This can be seen in the Fig. 16, when the complementary inputs are at logic '0', the two outputs do not discharge to the ground potential. On the other hand, if the complementary inputs are both logic '1', then the complementary output nodes will be charged through the pMOS transistors, see Fig. 2, which follows the power-clock. At the same instant the nMOS transistors (connected to ground potential) will discharg the output nodes to ground, settling the two output nodes at some intermediate value. These invalid conditions are depicted in Fig. 13 (b) and Fig. 16.

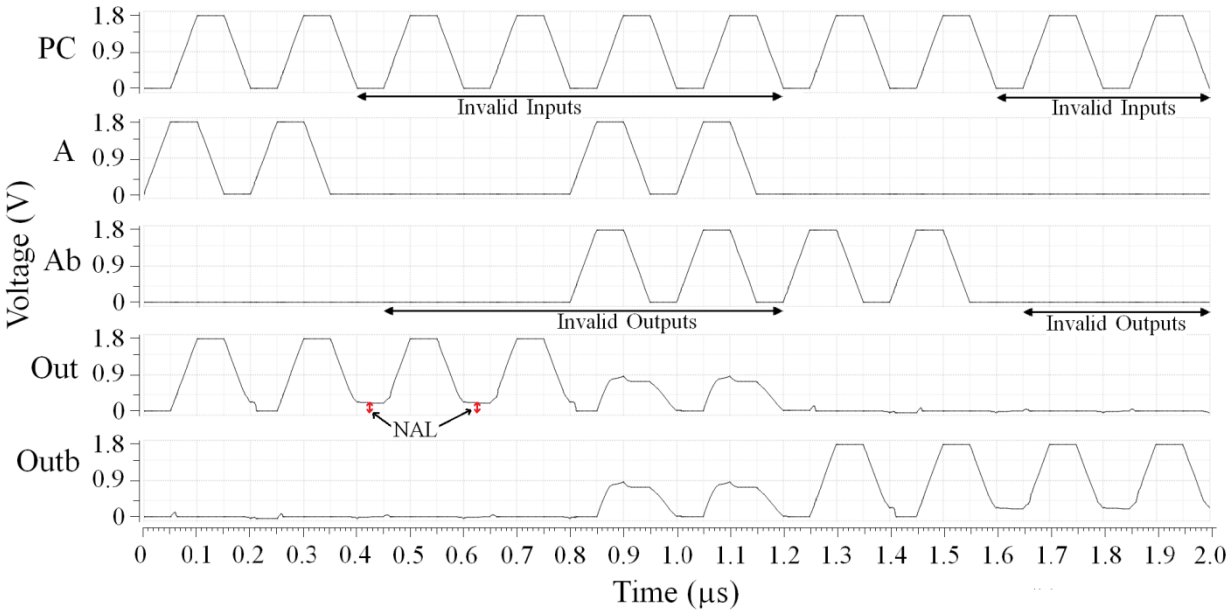


Fig. 16. SPICE waveform results for PFAL NOT/BUF gate with invalid outputs

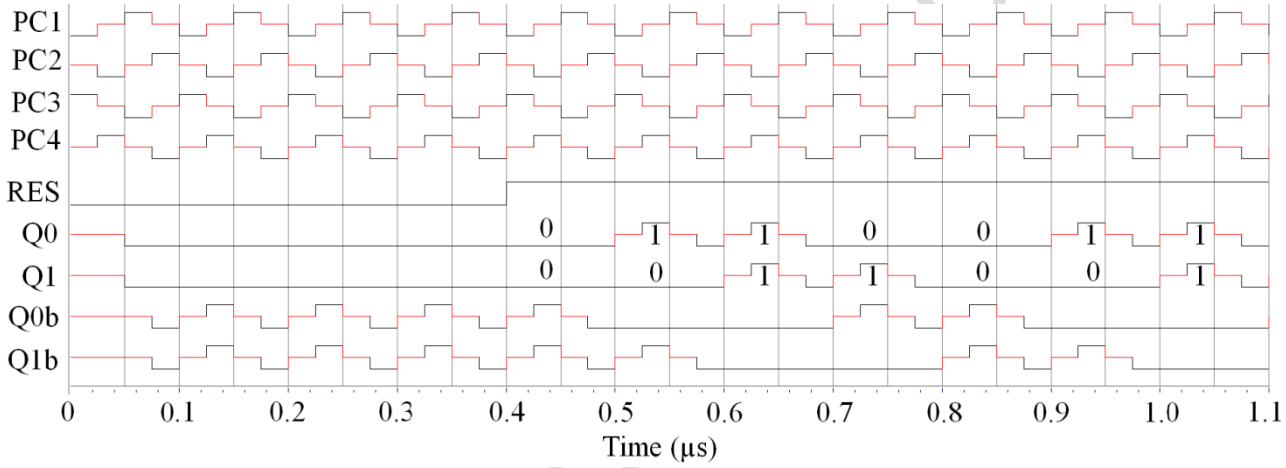
Our proposed VHDL model can easily identify the errors caused by the complementary inputs when both are at logic '0', which the SPICE simulation fails to identify. In this case, the circuit remains inactive, that is, the complementary output voltages remain at logic '0', detecting an invalid input condition. Similarly, when the complementary inputs are at logic '1', the complementary output nodes remain at a high impedance state encoded by 'Z'. The above invalid operations are shown in Fig. 13(b). Thus, two different states, '0' and 'Z' are used when the complementary inputs are driven to the same logic values (invalid conditions). This helps in identifying the value of the invalid inputs precisely.

5. Simulation Results

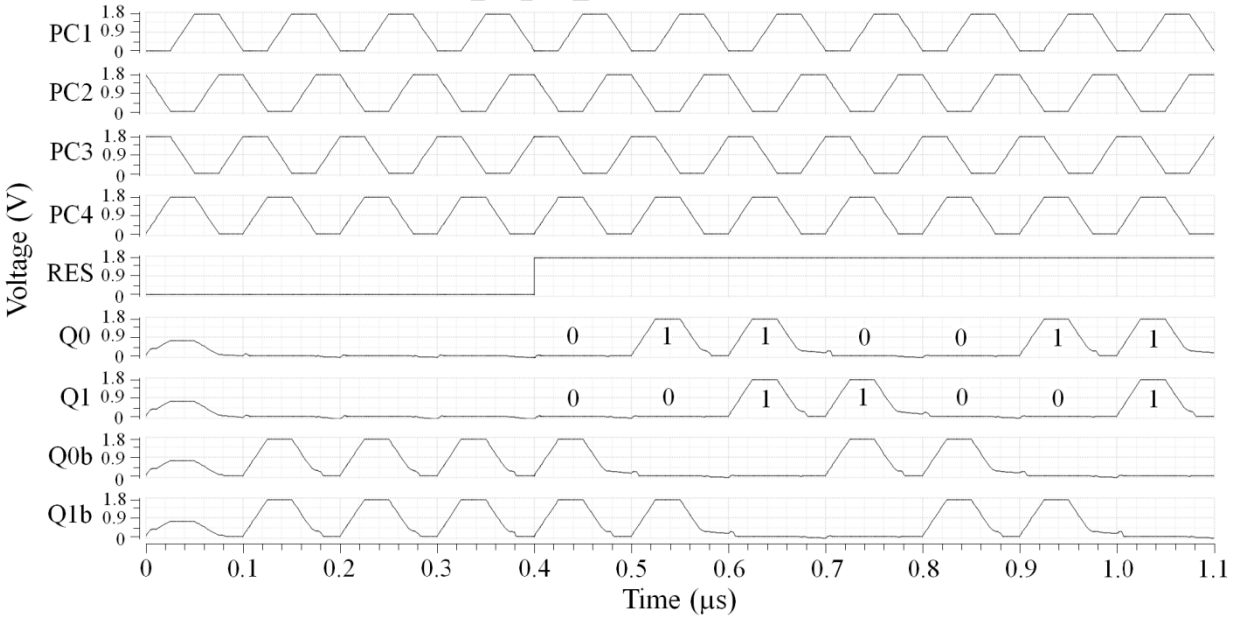
The 4-phase adiabatic logic family used for the SPICE simulation is PFAL. The transistor sizes are set to the technology minimum (Wmin=Wn=Wp=220nm, Lmin=Ln=Lp=180nm). The SPICE simulations were performed with the Spectre simulator using the Cadence EDA tool in a 'typical-typical', TT process corner using TSMC 180nm CMOS process at 1.8V power supply. The simulation is performed at 10MHz, chosen as a round number close to the 13.56MHz smart card center frequency. For all the other adiabatic logic gates such as AND/NAND, OR/NOR, XOR/XNOR and MUX/DeMUX the VHDL behaviour is described by combining the functional part and the adiabatic NOT/BUF for timing validation.

Using our homegrown cell library, the structural models of a 2-bit ring counter and a 3-bit up-down counter were successfully verified. The circuit functionality and timing verifications were done using HDL Designer from Mentor Graphic. The time period of the power-clock was taken as 100ns, having equal time for the four periods of the power-clock i.e. 25ns each. The simulation setup for VHDL is similar to that of SPICE so that uniformity and comparability are maintained across both the simulations.

Since the twisted ring counter is able to self-initialize from an all-zeros state and does not have any external inputs, it is an ideal vehicle for comparing the VHDL implementation with the



(a)



(b)

Fig. 17. 2-bit ring counter output waveforms (a) VHDL Model simulation (b) SPICE simulation

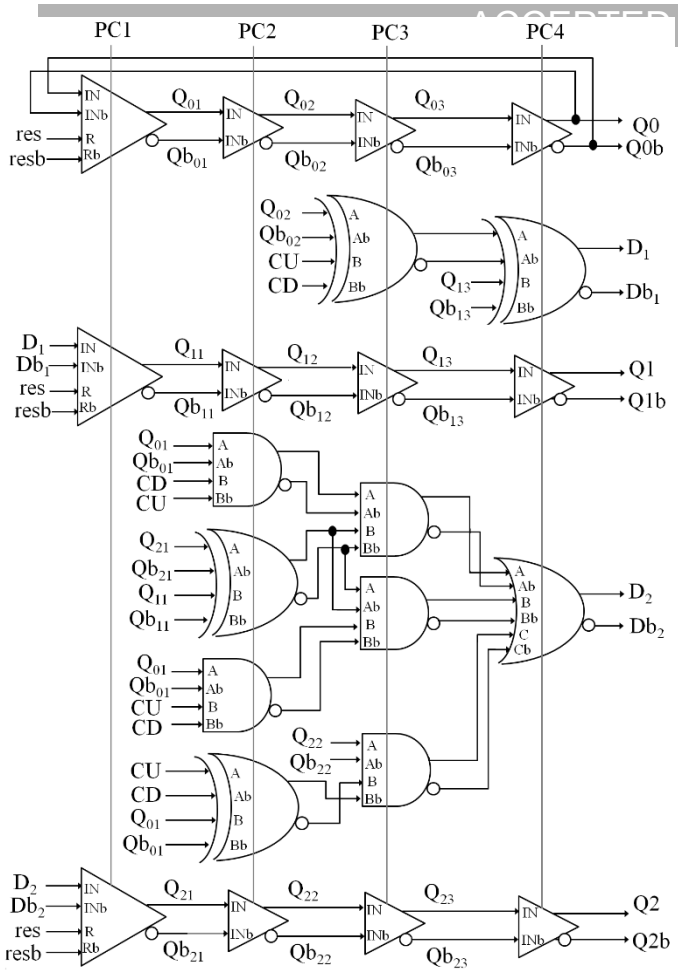


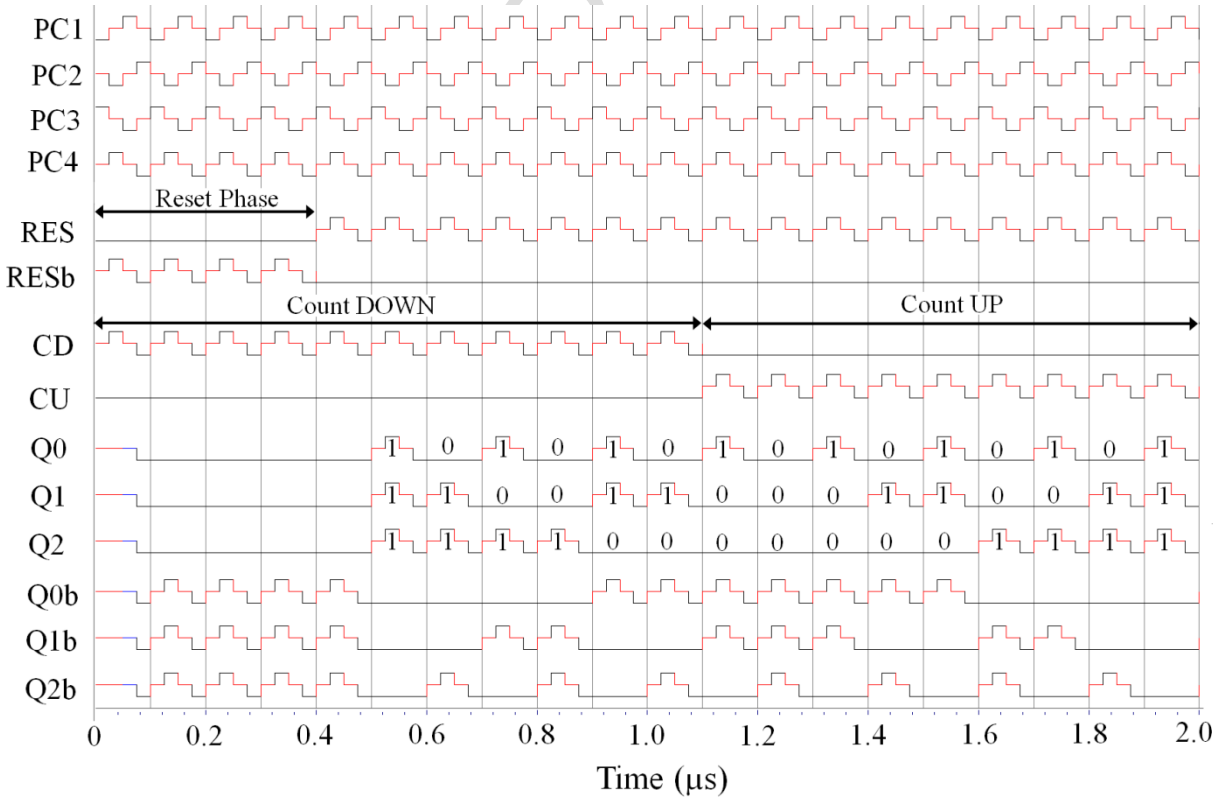
Fig. 18. Circuit diagram of a dual-rail adiabatic designed 3-bit up-down counter.

SPICE simulation. The 2-bit adiabatic twisted ring counter consisting of two D flip-flops [28]. The complete design of the 4-phase 2-bit ring counter is given in [28]. The reset input is given as a step signal, not converted to multi-level adiabatic input. Fig. 17 (a) and (b) shows the two output waveforms, one generated using the VHDL simulation and the other using the SPICE simulation respectively. As can be seen from Fig. 17 (a) and (b), there is no variation in the timing of the VHDL simulation in comparison to that of the circuit simulation waveform of the ring counter

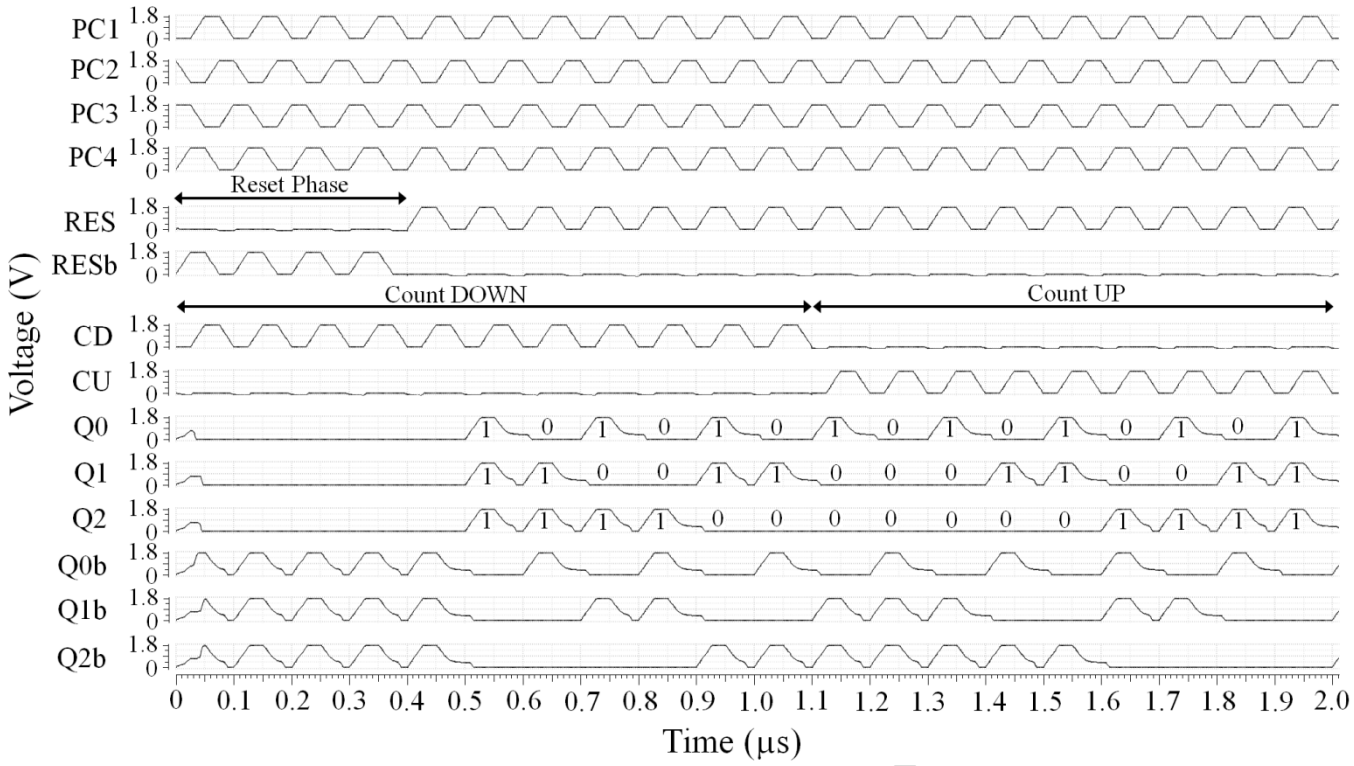
As the ring counter does not employ any combinational logic, a 3-bit up-down counter is modelled using HDL. The dual-rail reset signals are coded and converted to the multi-level adiabatic inputs (RES and RESb). The counter starts counting only when the reset signal 'RES' is high. Depending on the 'CD' and 'CU' signals, the counter either count down or up. The boolean expression for the 3-bit up-down counter is given in [30] and its corresponding circuit diagram is given in Fig. 18. The VHDL simulation waveform alongside the SPICE simulation for the 3-bit up-down counter is shown in Fig. 19 (a) and (b) respectively. The counter design reveals the accuracy of the modelling in terms of timing and representation of the adiabatic logic technique

6. Conclusion

VHDL provides a fast methodology to reduce the amount of time required for synchronizing and detecting design errors at an early design phase. This paper presents a new VHDL-based modelling approach for the 4-phase adiabatic logic circuits through the implementation of the 2-bit ring counter and 3-bit up-down counter. The exact behaviour of the trapezoidal power-clock is represented by presenting all the four periods distinctively using VHDL using the proposed approach. The shortcomings of the existing approaches especially of using a



(a)



(b)

Fig. 19. 3-bit up-down counter output waveforms (a) VHDL Model simulation (b) SPICE simulation

square wave as a power-clock is demonstrated. The simulation results of the NOT/BUF PFAL gate shows that if either the power-clock or the input signal is delayed or arrives early then the existing approaches malfunction and violates the adiabatic principle. Whereas, our proposed approach works functionally correct and obeys the adiabatic principle. The proposed approach further shows the precise timing mapping which is in agreement with that of the SPICE circuit level simulations. The simulation results of the two design examples of the counter circuits confirms the validity of our proposed modelling and hence can be used for the design and validation of a large complex 4-phase adiabatic system.

ACKNOWLEDGMENT

The authors wish to thank the University of Westminster for awarding the Cavendish Research Scholarship to carry out research in the Department of Engineering.

REFERENCES

[1] V. S. K. Bhaaskaran, "Energy recovery performance of quasi-adiabatic circuits using lower technology nodes," International Conference on Power Electronics, (2011), 1-7.

[2] M. Chanda, S. Jain, S. De and C. K. Sarkar, "Implementation of Subthreshold Adiabatic Logic for Ultralow-Power Application," IEEE Transactions on VLSI Systems, 23(12), 2015, 2782-2790.

[3] G. Pillonnet, H. Fanet and S. Houri, "Adiabatic capacitive logic: A paradigm for low-power logic," IEEE International Symposium on Circuits and Systems (ISCAS), (2017), 1-4.

[4] A. Galisultanov, Y. Perrin, H. Fanet, L. Hutin and G. Pillonnet, "Compact MEMS modeling to design full adder in Capacitive Adiabatic Logic," 48th European Solid-State Device Research Conference (ESSDERC), 2018, 174-177.

[5] S. Houri, G. Billiot, M. Belleville, A. Valentian and H. Fanet, "Limits of CMOS Technology and Interest of NEMS Relays for Adiabatic Logic Applications," IEEE Transactions on Circuits and Systems I: Regular Papers, 62(6), (2015), 1546-1554.

[6] M. Vollmer, and J. Götze, "An adiabatic architecture for linear signal processing", Advances in Radio Science, (2005), 325–329.

[7] L. Varga, G. Hosszu and F. Kovacs, "Two-level Pipeline Scheduling of Adiabatic Logic," 29th International Spring Seminar on Electronics Technology, (2006), 390-394.

[8] David John WILLINGHAM, "Asynchrobatic logic for low-power VLSI design," PhD. Thesis, 2010.

[9] R. Tessier, D. Jasinski, A. Maheshwari, A. Natarajan, Weifeng Xu and W. Burleson, "An energy-aware active smart card" IEEE Transactions on VLSI Systems, 13 (10), (2005) 1190-1199.

[10] E. Amirante, A. Bargagli-Stoffi, J. Fischer, G. Iannaccone and D. Schmitt-Landsiedel, "Variations of the Power Dissipation in Adiabatic Logic Gates," International Workshop on Power and Timing Modelling, Optimization and Simulation (PATMOS), (2001), 9.1.1–9.1.10.

[11] J. G. Koller and W. C. Athas, Adiabatic Switching, Low Energy Computing, and the Physics of Storing and Erasing Information, 2nd Workshop on Physics and Computation, (1992) 267-270.

[12] W. C. Athas, L. J. Svensson, J. G. Koller, N. Tzartzanis and E. Ying-Chin Chou, "Low-Power Digital Systems Based on Adiabatic switching Principles", IEEE Transactions on VLSI Systems, 2(4), (1994) 398-407.

[13] John S. Denker, A Review of Adiabatic Computing, IEEE Symposium Low Power Design, (1994) 94-97.

[14] Himadri Singh Raghav, V. A. Bartlett, and I. Kale, "Investigation of stepwise charging circuits for power-clock generation in Adiabatic Logic", 12th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME), (2016), 1-4.

[15] Himadri Singh Raghav, V.A.Bartlett and Izzet Kale, "Energy efficiency of 2-step charging power-clock for adiabatic logic", International Workshop on Power and Timing Modelling, Optimisation and Simulation (PATMOS), (2016), 176-182.

[16] L. Svensson, J. Koller, "Driving a capacitive load without discharging fCV^2", Proc. of the IEEE Symposium on Low-power Electronics, (1994), pp. 100–103.

[17] Athas, W. C., Koller, J. G., and Svensson, L, "An Energy-Efficient CMOS Line Driver Using Adiabatic Switching", Proc. of IEEE Great Lakes Symposium on VLSI, 1994.

[18] D. Maksimoviü and V. G. Oklobdžija, "Integrated Power Clock Generators for Low Energy Logic", 26th Annual IEEE Power Electronics Specialists Conference (PESC'95), 1, (1995) 61-67.

[19] N. Jeanniot, A. Todri-Sanial, P. Nouet, G. Pillonnet and H. Fanet, "Investigation of the power-clock network impact on adiabatic logic," IEEE 20th Workshop on Signal and Power Integrity (SPI), (2016), 1-4.

[20] N. Jeanniot, G. Pillonnet, P. Nouet, N. Azemard and A. Todri-Sanial, "Synchronised 4-Phase Resonant Power Clock Supply for Energy Efficient Adiabatic Logic," *2017 IEEE International Conference on Rebooting Computing (ICRC)*, Washington, DC, 2017, pp. 1-6.

[21] D. Maksimovic, and V. G. Oklobd˘zija, "Clocked CMOS adiabatic logic with AC power supply" Solid State Circuit Conference,(1995) 370-373.

[22] Jianping, H., Lizhang, C., and Xiao, L., "A New Type of Low-Power Adiabatic Circuit with Complementary Pass-transistor Logic" Proc. 5th International Conference on ASIC, 2, (2003) 1235–1238.

[23] Sachin Maheshwari, V.A.Bartlett and Izzet Kale "Energy efficient implementation of multi-phase quasi-adiabatic Cyclic Redundancy Check in near field communication", Integration, The VLSI Journal, Elsevier, 62, (2018), 341-352.

[24] A. Vetuli, S.D. Pascoli and L.M. Reyneri, "Positive Feedback in Adiabatic Logic" Electronics Letters, 32(20), (1996) 1867-1869.

[25] F. Liu and K. T. Lau, "Improved Structure for Efficient Charge Recovery Logic", Electronics Letters, 34(18), (1998),1731-1732.

[26] L. Varga, F. Kovacs, & G. Hosszu, "An efficient adiabatic charge-recovery logic" SoutheastCon 2001. Proceedings. IEEE, Clemson, SC, (2001) 17–20.

[27] Y. Moon and D.K. Jeong, An Efficient Charge-Recovery Logic. IEEE Journal of Solid-State Circuits, 31(4), (1996) 514-522.

[28] Sachin Maheshwari, Viv A. Bartlett, I. Kale, "4-phase resettable quasi-adiabatic flip-flops and sequential circuit design" 12th Conference on PhD. Research in Microelectronics and Electronics (PRIME), (2016), 1-4.

[29] M.C. Knapp, P. J. Kindlmann, M. C. Papaefthymiou, "Design and Evaluating of Adiabatic Arithmetic Units", Analog Integrated Circuits and Signal Processing, 14, (1997) 71-79.

[30] Sachin Maheshwari, V.A.Bartlett and Izzet Kale, "Adiabatic Flip-Flops and Sequential Circuit Designs using Novel Resettable Adiabatic Buffers" 23rd European Conference on Circuit Theory and Design (ECCTD), (2017) 1-4.

[31] C. O. Campos-Aguillón, R. Celis-Cordova, I. K. Hänninen, C. S. Lent, A. O. Orlov and G. L. Snider, "A Mini-MIPS microprocessor for adiabatic computing," *IEEE International Conference on Rebooting Computing (ICRC)*, (2016), 1-7.

[32] Ph. Teichmann, J. Fischer, F. Chouard, and D. Schmitt-Landsiedel, "Design Issues of Arithmetic Structures in Adiabatic Logic" Adv. Radio Science, 5, (2007) 291-295.

[33] P. Teichmann, Adiabatic logic: Future Trend and System Level Perspective, Springer Science & Business Media, 34, 2011, 10-14.

Sachin Maheshwari received the BTech Degree in electrical and electronics Engineering 2007 from ICFAI Tech, Hyderabad, India and Master of Engineering Degree in Microelectronics from Birla Institute of Technology and Science (BITS), Pilani, India in 2009. Currently, he is a PhD student at the University of Westminster, London, UK in Applied DSP and VLSI Research Group, Department of Engineering.
His research interest is in Low Power Digital VLSI Design.

Viv Bartlett obtained his BScEng (Hons) at Imperial College, London and his PhD at the University of Westminster. He is currently a Principal Lecturer at the University of Westminster, a post he has held for the last 10 years, teaching mainly in the fields of Digital Systems and VLSI design.
He has been designing integrated circuits since the 1980s and his research interests are centred on VLSI design and DSP architectures with an emphasis on asynchronous and low-power techniques.

Izzet Kale (M'87) received the B.Sc. (Hons.) degree in electrical and electronic engineering from the Polytechnic of Central London, the M.Sc. degree in the design and manufacture of microelectronic systems from Edinburgh University, Scotland, and the PhD degree in techniques for reducing digital filter complexity from the University of Westminster, London. He is currently a Professor of applied DSP and VLSI systems, and Head of Engineering and Founder and the Director of the Applied DSP and VLSI Research Group, University of Westminster. He is currently working on efficiently implementable, ultralow-power DSP algorithms/architectures, sigma-delta modulator structures and Low Power VLSI Design for secure systems.

Highlights

- A new approach for modelling 4-phase adiabatic logic circuits using VHDL is presented
- The proposed approach includes the modelling of the dual-rail input and output signals.
- Shortcomings of the existing HDL-based approaches are discussed.
- The exact behaviour of the trapezoidal power-clock is represented by modelling all the four periods distinctively.
- The functional aspects of the proposed approach are verified for a 2-bit ring counter and a 3-bit up-down counter.