Challenges to adopting adiabatic circuits for systems-on-a-chip

Krishnan S. Rengarajan | Saroj Mondal | Ravindra Kapre

Abstract
Adiabatic complementary metal–oxide–semiconductor (CMOS) circuits have been proposed as a low-power option for CMOS systems-on-a-chip (SoCs) but have not gained popularity due to practical difficulties in scaling to millions of gates. The architecture of a pipeline of stages with slow-transitioning clock phases demands the generation and distribution of clock phases precisely and efficiently. This power must be more than offset by the power saved by using adiabatic circuits. The problems in adiabatic logic circuits are described, and solutions are proposed to address them. Three published topologies are considered, namely positive-feedback adiabatic logic (PFAL), two-level adiabatic logic (2-LAL) and clocked adiabatic logic in 40 nm CMOS technology at 100 MHz. New circuit ideas for complete level restore in PFAL and avoidance of floating nodes in 2-LAL are presented. The problem with 2-LAL multi-input gates is published and solved for the first time here using a modified PFAL. The conclusion is that a 3X power savings in PFAL is about the best that can be achieved in an SoC context—a low return given the required investments in area and complexity. This should motivate the future discovery of more efficient solutions.

1 INTRODUCTION
Adiabatic computing has been investigated as a path to low-power complementary metal–oxide–semiconductor (CMOS) chips [1–8]. The literature shows the benefit of adiabatic circuits at the level of a single gate or a small, structured ensemble of gates organised conveniently for multiphase clocking. There are also reports of chips designed using adiabatic circuits [8]. However, these circuits have drawbacks such as floating nodes, poor efficiency at higher frequencies, and narrow scope of application due to complex clocking schemes [9]. Generally, there is little description of the challenges involved in combining gates. However, [10] discusses the termination problem in adiabatic styles such as split-level charge recovery logic (SCRL) [5], reversible energy recovery logic (RERL) [11,12] and two-level adiabatic logic (2-LAL) [13]. This was our finding as well, and as described later, it is a different experience category that seems more advantageous for power and area—e.g. 2-LAL vs. the positive-feedback adiabatic logic (PFAL) buffer/inverter. We shall discuss the circuit issues with these categories. The promise of the second category is invalidated because of these issues. We also combine first-category principles to solve problems in the second category. The circuits we consider fall into one of the above two categories. For example, efficient charge recovery logic (ECRL) [14] and modified ECRL are adiabatic logic styles that can be broadly classified into two categories: (a) styles such as PFAL [1,6] and clocked adiabatic logic (CAL) [8] where energy recovery is contained in the unit cell (such as inverter, Nand) (b) styles such as SCRL [4], 2-LAL [13], and RERL [11,12] where the outputs are recovered by the next stage. Secured quasi-adiabatic logic (SQAL) falls into the first category due to its similarities with PFAL. Similarly, symmetric pass gate adiabatic logic (SPGAL) is a slight modification of PFAL that uses discharge control when inputs are set [16].

In addition, circuit issues, the primary difficulties of multiphase clock generation and implementation of CMOS-level compatible random logic (as opposed to solely regular structures like multipliers [1] and adders [17]—these implementations call for eight clock phases) without compromises in performance must be overcome to adopt adiabatic logic for systems-on-a-chip (SoCs). Table 1 classifies the adiabatic styles as Categories 1 and 2. Circuits like SAL [20] and those that use diodes (e.g. 2-2N2D, ADL, IAPDL, DAPDL [20], and QSRL...
TABLE 1 Adiabatic logic styles shortlist for system-on-a-chip applications

| Adiabatic circuit                      | Category |
|---------------------------------------|----------|
| Efficient charge-recovery logic (ECRL) | 1        |
| Positive-feedback adiabatic logic (PFAL) | 1        |
| Symmetric pass gate adiabatic logic (SPGAL) | 1        |
| Clocked adiabatic logic (CAL) | 2        |
| Pass-transistor adiabatic logic (PAL) | 2        |
| 2N2P2NP | 2        |
| Secured quasi-adiabatic logic (SQAL) | 2        |
| Split-level charge-recovery logic (SCRL) | 2        |
| Two-level adiabatic logic (2-LAL) | 2        |
| Reversible energy-recovery logic (RERL) | 2        |

In practice, the charge dissipation for each transition is half $V_{DD} \times C \times T$ joules from supply—dissipation near zero. The rules for designing adiabatic circuits are described in [13].

The key idea of the logic styles evaluated in this paper is to set the control input (e.g., gate) of the switch (MOSFET) to an ON state only when the output node (say drain) and the clock/supply node (say source) are at the same level and not otherwise. This allows the drain to follow the source without a voltage drop across the switch during evaluation. The different states of each switch are managed through the definition of different clock phases.

Charged or floating nodes whose states define logic operations are a concern for a robust chip architecture, although simulations may not adequately show the effect of leakage at operating frequencies. Non-adiabatic switching losses ($|V_{DS}| > 0$) must be managed with considerable attention because they can easily wipe out savings from adiabatic operation. Some examples are (a) the need for the supply/ground clock phase to reach a threshold voltage ($V_T$) of a fully turned-on (gate-asserted) MOSFET before it can conduct, (b) crowbar currents in cross-coupled differential structures while stabilizing the logic states, (c) the finite rise/fall times of the clock signals operating the adiabatic circuit as opposed to an infinite rise/fall time requirement for zero loss, (d) leakage loss, (e) incomplete charge recovery due to use of PFETs to pass LOW clock levels or n-channel field-effect transistors to conduct HIGH clock levels.

All of these effects can quickly weaken the case for adiabatic circuits when non-ideal effects are accounted for: device mismatch in differential circuits, VT variation through chain stages and clock-phase uncertainty due to variation from clock generation and distribution, non-ideal (not trapezoidal) clock wave-shape, clock timing accuracy due to supply/ground noise, crosstalk, IR drop and difference in IR drop between source and end points, degradation from negative-bias temperature instability and hot-electron etc. Also, the comparison against conventional CMOS styles must account for the same throughput (MHz), single-ended conventional CMOS versus the differential nature of adiabatic styles, ability to support gates with multiple inputs in different states, flexible arrival times of input signals to a gate versus precise setup for adiabatic operation. The last metric imposes the severe constraint that adiabatic logic must be strictly pipelined with each gate considered a pipeline stage. Routeability is often a challenge in SoCs today.

Multiple clock phases and increased cell areas pose considerable problems. There is invariably a need to interface with conventional full-swing CMOS circuits at the input side of adiabatic domains in an SoC. The adiabatic styles discussed here need complementary inputs that are in phase for maximum throughput (if not in phase, a setup margin needs to be budgeted). The complementary inputs introduce a non-adiabatic loss penalty as illustrated in the next section.

Unit cells are important for making larger blocks. However, in addition to logic function and low power, competitive area and performance must be delivered. [2] is a standard-cell-based simplified processor implementation, but the use of SCRL [4] is not easily adaptable to regular SoCs. Managing the clock signalling on supply and the ground rails for circuit function with precise timing across stages is a costly proposition for SoCs that contain millions of gates. Erik P D et al. [3] discuss nanodevices with high-level functions that aggregate several
logic functions and memory as the future path to low-power chips; however, these still need further research for practical implementations. Xuchu Hu et al. [24] discuss the implementation and optimisation of clock trees for SoCs using resonant clocking. This holds promise for clocking multiphase adiabatic systems provided that significant power savings for adiabatic logic can be demonstrated. Section 2 describes the three adiabatic logic styles PFAL [5], two-level adiabatic logic (2-LAL) [6], and CAL [7], highlighting the problems in an SoC context. We discuss practical difficulties, their possible solutions and cost. The message on power and area impact is clear from the data—the authors feel that a delineation of the problems and possible approaches to solve them is more valuable for the industry than publishing more accurate data from layout/silicon implementation or data on larger circuit ensembles. The circuit ideas for the problems of PFAL and 2-LAL, presented here for the first time, should motivate future work on more efficient solutions. Section 3 compares results against expectations for an SoC. Section 4 summarises our findings and discusses future work.

2 EVALUATION OF ADIABATIC CIRCUIT STYLES

We consider the three adiabatic circuit styles from the literature: PFAL, CAL and 2-LAL. After a brief review of circuit styles, we present simulation data and analysis for each of the three logic styles. We point out challenges, possible solutions and their cost as appropriate. It is worth noting upfront that in all these circuit styles, the logic function of the cell (e.g. inverter output X is inverted level of A) is valid only during the evaluation (E or e) phase. The input would be reset soon after, while the output will be held (H or h) at the evaluated value so that the next stage may be evaluated. Therefore, the differential outputs of these circuits are truly in opposite states only during the Hold phase. This rule does not apply at the interface of conventional CMOS and adiabatic domains where the former domain's outputs are complementary as discussed in the last section. Also, these signals will be illustrated as slow ramping like the adiabatic signals for ease of understanding. In reality, these are sharp edges and hence cause non-adiabatic loss (explained shortly) during the hold and restore phases of the first-stage outputs even more than illustrated here. In this paper, we will assume that clock phases are generated precisely and distributed across the SoC (e.g. using resonant clocking [24]). All stages are nominally loaded with 2fF of load (pCloud is a variable parameter that we will refer to as Cloud in this paper) at their outputs to represent interconnection and some fanout unless otherwise stated. All simulations use 100 MHz clocks. Basic gates including buffer/inverter, And/Nand, Or/Nor and the two-to-one multiplexer and logic chains that contain these are evaluated. We present the first three gate types and logic chains.

While comparing adiabatic power with conventional CMOS logic power, we compare for the same data throughput throughout a stage and consider all logic states of the logic function. The averaging window is for four input transitions (four alternate rising and falling transitions for buffer/inverter and the four binary input combinations for And/Nand or Or/Nor) at 100 MHz. This corresponds to four evaluation cycles for the first stage of an adiabatic logic chain. Note that for conventional full-swing CMOS, the reported power is for a single-ended signalling. The complementary input is a requirement of the functioning of adiabatic circuits. Adiabatic power is reported for a stage past the first stage, thus avoiding the interface issue.

2.1 Positive-feedback adiabatic logic

The PFAL buffer/inverter gate of Figure 1 [1] is described first. This style uses a clocked supply while the ground is static.

![Figure 1](image)

**FIGURE 1** Positive-feedback adiabatic logic buffer/inverter (a) Unit cell schematic (b) A chain of unit cells; outputs of the first stage are used to drive the logic in Figure 4b. (c) Simulation waveforms of the chain in (b); E = evaluation, H = hold, R = reset, I = setup inputs. The suffixes to these notations represent the stage number in the logic chain (suffixes are used in signal names, too); I/E/H/R (i/e/h/r) are used for stages 1,3 (2,4); A is the input assisted abrupt restore; B is the crowbar spike from non-adiabatic disturbances at the interface with conventional differential complementary metal-oxide-semiconductor data inputs; C is the incomplete level restore due to the PFET going off at zero overdrive; D is the full restore at start of evaluation; F is the non-adiabatic loss during the initial part of evaluation (due to non-zero voltage between drain and source, or VDS).
The practical problems of PFAL are better explained in the context of a logic chain (Figure 1b). Figure 1c shows the four quadrature phases of the clock that are used as supplies for subsequent stages: Phi1 for the first stage, Phi0N for the second stage, Phi1N for the third stage and Phi0 for the fourth stage (Figure 1b). This sequence repeats. The previous and next stages in a chain of gates are of the same topology as shown in Figure 1a but operate with one phase earlier and later, respectively—the evaluation (E/e) of one stage will be the input setup phase (I/i) of the next, while its holding phase (H or h) will coincide with both the evaluation of the next stage and the reset/recovery/restore phase (R/r) of the previous stage’s output. The letters I/E/H/R (i/e/h/r) are used for the input setup phase (I/i) of the next, while its holding phase (H or h) of the previous stage’s output. The letters I/E/H/R (i/e/h/r) are used for stages 1 and 3 (2 and 4).

In Figure 1a, nodes XB (buffered output, same polarity as Input A during E/e) and X (Inverted output, same polarity as Input AB during E/e) would be in reset (logic low) state before the start of evaluation (E/e)—see Stage1 waveforms in the 10–12.5 ns range in Figure 1c; IN and INN are the inputs to Stage1, and OUT1pfal and OUTN1pfal are its outputs as shown in Figure 1b.

Eventually, the cross-coupled amplifier core—MN1, MP1, MN2 and MP2 in Figure 1a—will hold the ‘LOW’ output at LOW when the inputs reset soon after evaluation. The input NMOS transistors MND1 and MND2 are in parallel with MP2, MP1, respectively. Input A (IN in Figure 1b) turns ON the gate of MND1 by going high (note that the supply-clock Phi1 is LOW currently), and Input AB (INN in Figure 1b) keeps the gate of MND2 OFF because it is LOW.

This is different from the case when the input is from an adiabatic stage (as for other stages in the chain) where it would have stayed LOW (reset state) rather than going from high to low—both inputs would have been low in their reset state, and only the input that must go high would do so. Note that Phi1 is low and so are XB (OUT1pfal in Figure 1b) and X (OUTN1pfal in Figure 1b) at this time. With IN and INN set as High and Low, respectively, during the I phase of the first stage, XB is all set to rise High when Phi1 rises during the next phase—evaluation.

Inputs IN and INN to the first stage in Figure 1b should be held stable (equivalent to H of a previous stage in PFAL parlance) during the E phase of the first stage when the clock Phi1 goes high very slowly—ideally, much slower than the RC time constant of the MOSFET switch and its capacitive loading so that XB faithfully follows Phi1 (evaluation) while avoiding a voltage drop (VSD = 0) across the switch MP2 during the transition of output XB. As Phi1 rises, the cross-coupling action through MN1 holds X low, allowing XB to pull high—this also turns ON MP2, which helps the tracking of XB with Phi1 when Phi1 has risen VT above X; the initial tracking of the XB node with Phi1 is through MND1. However, as the evaluation phase begins and MND1 passes Phi1 to XB (rising but still very low), the leakage increase in MND2 and MP1 drains charge off of XB, which MN1 fights against, as illustrated by the annotation ‘F’ in Figure 1c (OUT1pfal rising slightly at around 23 ns). Once Phi1 rises high enough to pull XB higher, MN1 turns on harder and MP1 goes off, providing a stable condition for XB and X. This crowbar during the initial evaluation is a loss that extends for a long time due to the slow ramping of the supply clock. In conventional CMOS circuits, this duration is very short due to fast switching inputs.

Further, at low voltages, depending on the VTs of MND1 and MP2, there can be a middle region during the transition of Phi1 when MND1 saturates while MP2 has still not turned on. This can happen for a low-supply voltage when VTs are high (e.g. slow process, cold temperature). A low-transistor VT is favourable to overcome this problem, but it increases crowbar and leakage currents. Once evaluated, energy is stored at node XB. The evaluated outputs are then held (H1 in Figure 1b) static while the subsequent stage (stage2) evaluates—during this phase (e2 in Figure 1c), a change in inputs IN, INN is immaterial to the evaluated value of the output because XB and X are latched by the cross-coupled structure. Figure 1c shows IN going back to Low consistent with the recovery phase assumption of adiabatic logic. However, INN also goes high in keeping with its complementary nature and remains high during the next phase. Thus, even during the H1 phase, there is crowbar due to INN trying to pull X high through MND2 against MN1 (annotation ‘B’ in Figure 1c). This crowbar component can be very high and can increase the average power consumption by a few orders of magnitude (e.g. nA to uA).

While comparing the power of various styles later, we consider only the purely adiabatic context, that is, that inputs come from adiabatic circuits. Subsequent to the H phase is the charge-recovery phase, R. For an adiabatic stage, unlike in conventional CMOS circuits, the charge is returned to the supply clock (e.g. replenishing decoupling capacitors). When the supply clock goes low, XB tracks it—the charge is returned to the supply for future use—to the point when the PFET MP1 stops conducting, leaving XB at a level not fully low. This represents another non-adiabatic loss in charge recovery in PFAL (annotation ‘C’ in Figure 1c for stage3). The node stays charged at this level (or settled as a function of leakage to ground) until the next evaluation when, depending on the data, it could reach a full LOW (annotation ‘D’ in Figure 1b for stage 4). In addition, for the first stage, with INN being high (MND2 ON, so MN2 ON), the reset of XB is accelerated as shown in annotation ‘A’ in Figure 1c. This again amounts to non-adiabatic loss. The problems of annotations ‘A’ and ‘B’ can be avoided if we use NMOS isolation transistors (controlled by Phi0) stacked in series with MND1 and MND2. Phi0 goes LOW during H1 (the crowbar problem occurs during the latter half of H1, and Phi0 is sufficiently low to isolate inputs from outputs) and is stable LOW during R1. This requires doubling of MND1 and MND2 (the stacked NMOS will be the same size). This means more input capacitance (also extra intermediate nodes and more output parasitic capacitance) causing increases in power. The area will increase as well.

Note that the bodies of PMOS FETs are best connected to constant supply levels rather than to clocked VDD to contain the area increase due to guard rings. Similarly, the bodies of NMOS FETs may be connected to ground. Low-VT FETs may be used to make up for the loss in performance from
body-biasing (note that VDD clock swings may be lower than full swing, Figure 1b shows the swing is supply-2 * 100 mV: between 1.26 and 0.1 = 1.16 V, and 0.1 V). Table 2 shows the effect of body-biasing on power from simulations at 100 MHz, TT, 1.26 V, 125°C for a buffer/inverter stage. When Nwell is shorted to the clocked VDD, the VT of the device is lower (lack of body bias), and the output tracks the supply change well. The energy recovered also decreases when Nwell is made constant—the output recovery is to a higher level (PFET turns off sooner due to higher VT)—280 mV versus 435 mV. Use of lower-VT FETs (RVT instead of HVT) helps improve the power. A trapezoidal power clock waveform returns the best power for adiabatic circuits [8]. We will use the lower-VT FETs for comparisons henceforth. The use of clocked VDD for Nwell is a challenge in an SoC that consists of significant IP made of conventional CMOS; however, we consider this here with an assumption of deep Nwell-isolated islands for other hard-IP. From Table 2 (last row = 0.125), it can be seen that there is an 8X benefit at TT, 1.26 V, 125°C, 100 MHz, and 2 fF loading. The benefit decreases to ~6.5X at VDD = 1.0 V (Figure 2b), and further to ~3X at 25°C (Figure 2c).

It is useful to understand the power benefit as a function of frequency, output load, temperature and supply voltage. This information is shown in the charts of Figure 2 and is self-explanatory. It may be noted that the lower efficiency of adiabatic circuits at lower frequencies is due to the prolonged (slow ramp) occurrence of non-adiabatic losses and leakage during the E, H and R phases as explained earlier. The PFAL inverter/buffer power is about three times better than conventional CMOS at TT, 1.0 V, 25°C, 100 MHz. This is hardly sufficient when one considers operation at process, voltage, temperature extremes, and clocking power overhead due to multiple phases and additional area investments.

We now look at other logic functions and a chain consisting of mixed-gate types. Figure 3 shows the unit cells of And/Nand and Or/Nor gates.

The output X is the inverted function (Nand/Nor) and the output XB is the non-inverted (AND/OR; Figure 4). In Figure 4b, cursors V1, V2, V3 and V4 are placed right in the middle of the evaluation phases of Stage1, Stage2, Stage3 and Stage4, respectively.

The output X is the inverted function (Nand/Nor) and the output XB is the non-inverted (And/Or). In Figure 4b, cursors V1, V2, V3, and V4 are placed right in the middle of the evaluation phases of Stage1, Stage2, Stage3, and Stage4, respectively.

**Table 2**  
Nwell tie & field-effect transistor threshold voltage impact on power (100 MHz, TT, 1.26 V, 125°C)

| Circuit configuration | Ratio of PFAL to conventional CMOS power |
|-----------------------|-----------------------------------------|
| Nwell shorted to clocked VDD | 0.156 |
| Nwell held constant at supply VDD | 0.22 |
| Nwell held constant at supply VDD; lower-VT FETs | 0.125 |

Abbreviations: CMOS, complementary metal–oxide–semiconductor; FETs, field-effect transistors; PFAL, positive-feedback adiabatic logic; VDD, voltage drain; VT, threshold voltage.

**Figure 2**  
Ratio of adiabatic to conventional complementary metal–oxide–semiconductor (CMOS) power (lower is better) versus (a) Frequency (b) Voltage (c) Temperature (d) Output load
The signals are grouped as clock phases (in order of connection to stages 1–4): Phi0 = thick solid, Phi0N = thick dashed, Phi1 = thin solid, Phi1N = thin dashed; frequency = 100 MHz, non-inverted inputs and outputs (e.g. OUTor1pfal, solid) and inverted inputs and outputs (e.g. OUTnor1pfal, dashed) in that order. In Figure 4a, note that the stage1 outputs from Figure 1b, OUT1pfal and OUTN1pfal (E1 evaluation) are wired to the true and complement inputs of a buffer/inverter, respectively. This buffer/inverter works in parallel with the And/Nand stage, evaluating rising Phi0N. Its outputs, OUT2_Bpfal (evaluated HIGH in accordance with the OUTN1pfal connected to AB input of buffer/inverter) and OUT2_BNpfal are interchanged when feeding the next stage: OUT2_BNpfal to A2B input of Nor gate (stage3) and OUT2_Bpfal to A2 input.

The limitation that logic stages are pipelined and operating on a clock phase that is an intermediate output of one chain can be used as input to a receiving stage in another chain only if evaluation in the same phase as the stage preceding the receiving stage is obvious. This contrasts with conventional combinational logic where one could just wire up any stage's output to any stage's input downstream of another chain.

At the end of four cycles, we see the input data propagating after functional modifications through the stages to the output of stage 5. Cause and effect lines in Figure 4b, showing the function and timing during the evaluations of each stage, should help the reader to understand the operation of the mixed-logic PFAL chain. For example, stage 1 (Nor PFAL gate) evaluation is annotated as E1, where the inputs INA = 1 and INB = 0 at Phi1 rising result in OUTor1pfal = 1 and OUTnor1pfal = 0. Evaluating at Phi0N rising (E2) are two parallel stages—And/Nand with true (complement) inputs shorted and a buffer/inverter—and their results (OUTand2 = 0) feed into the third stage, a NOR gate. The NOR gate evaluates at E3, and its outputs feed a buffer/inverter that evaluates at E4. Its output in turn drives an And/Nand with inputs shorted, resulting in OUTand5pfal and OUTnand5pfal. Note that the second E1 evaluation (at ~24 ns) of the OUTor1pfal gate evaluates to a LOW corresponding to INA = 0, INB = 0.

A variation of PFAL discussed here, called 2N2N2P, is described in [19], where the MND transistors form a pull-down network. The operating principle and problems are similar. The conventional CMOS full swing, complementary, in-phase inputs INA, INAN, INB, and INBN feed into the first stage (NOR) with the associated interface problems discussed earlier for the buffer/inverter chain. The supply-clock swings are 1.06 V: 100 mV down from 1.26 V and 100 mV up from ground. We will compare the three styles for this configuration and stimulus, at the end of this section.
2.2 Clocked adiabatic logic

CAL [7] is like PFAL in the use of clocked supply. However, it has only one phase that connects to all stages, with every alternate stage evaluating in alternate slots, so it repeats at two times the frequency of a clock phase used in PFAL. However, keeping in mind the clock generation power overhead, it is best to keep the PCLK frequency constant at 100 MHz across the styles. The simulation is done with PCLK at 100 MHz. The power comparison is for the same throughput. Figure 5 shows the CAL unit cell and/nand. Other cells can be made by suitable modifications of the pull-down legs. The device sizes have increased from PFAL to achieve the same performance. It uses a single supply (VDD) trapezoidal clock, Pclk, for the evaluation (Pclk low to high).

Prior to Pclk going high, the inputs are set to the cross-coupled latch through the pull-down structures when enabled by a conventional clock Phi. (Phi and PhiN enable alternate stages). Phi and PhiN add extra non-adiabatic clocking power compared with PFAL or 2N2N2P. Figure 5b shows the timing waveforms of mixed-logic chain (same as Figure 4a but with unit cells replaced with the corresponding CAL unit cells. The inputs to the parallel second path buffer/inverter is from a CAL equivalent of Figure 1b—not redrawn here). The High state of Phi (PhiN) has no function other than an idle phase that is there is no Hold phase as in PFAL because the evaluated (during Phi rise) full swing values on the cross-coupled latch, cannot be used by the next stage that is disabled at this time. The incomplete restore problem discussed in PFAL applies here too. However, it is used as a functional feature here—the residual level present at the cross-coupled latch during the disabled state (Phi LOW) is supposed to flip the latch back to the previously evaluated value when Pclk rises again. Thus, the same latch value of the enabled cycle is repeated during the disabled cycle—a pair of identical pulses occurs on the cross-coupled latch. However, the disabled cycle, being the enable cycle of the next stage (PhiN High), uses this repeating value (as it settles) to evaluate its pull-down network that is the disabled stage and next stage that is enabled, evaluate simultaneously, such that the enabled next stage uses the value that is arriving from the disabled stage.

Note that the outputs of the parallel second stage buffer/inverter that are labelled OUT2_B and OUT2_BN, behave differently from that of Figure 4(b) of PFAL that is OUT2_Bpafal and OUT2_BNpafal. This parallel stage receives its inputs from the outputs of the first stage of Figure 1b (with PFAL unit cells replaced with CAL unit cells). This different behaviour of OUT2_B and OUT2_BN is because IN (INN is complement) was made HIGH and LOW synchronous with Phi. The first stage of Figure 1b (in this case, CAL version is used) now sees the same value on IN and INN inputs every time Phi goes HIGH (enable phase). During the subsequent disabled phase, it evaluates back HIGH from the residual differential value. Thus, OUT2_B evaluates to HIGH at every PCLK. OUT2_BN permanently remains LOW. The third stage (see Figure 4b) being an or/nor stage, responds with a HIGH to its HIGH input, every time. The subsequent stage being a buffer/inverter, responds similarly. Crowbar at the start of evaluation exists as discussed before. Adding a pullup structure like in PFAL removes this problem but increases input capacitance (adds power), hence not useful.

Duplicate evaluations (when enabled by Phi = High and then when disabled by Phi = low) and reset is substantial power as was borne out in the simulation results of a logic chain. While the idea of a single power clock is attractive, the higher speed supply-clock requirement is a downside considering clock distribution on a SoC. Importantly, it may be seen in Figure 5b that all the evaluations (rising transitions on the outputs of all stages) are non-adiabatic (sharper than in PFAL). This is because of the simultaneous settling of the input from the disabled previous stage while a stage is evaluating. Contrast this with PFAL (Figure 4b or Figure 1c), where the inputs are settled in the I phase, and the output transitions track the supply-clock phase. Slowing down the ramp is a possibility, but it will affect performance and timing margins at the 100 MHz target, hence not preferred.
2.3 Two-level adiabatic logic

2-LAL was originally proposed by M. P. Frank. [6] is one of many references on 2-LAL. Figure 6 shows the basic element of 2-LAL, which comprises two transmission gates in parallel, conveying true and complement inputs, A and AN, to the outputs, X and XN, respectively, under the control of P (to NMOS gate) and PN (to PMOS gate). The size of the FETs is important as it increases the capacitive load on the clocks PhiX and PhiXN (X = 0,1). Figure 6(b) shows the buffer/inverter implementation. IN and INN are the true and complement inputs, while OUT and OUTN are the true and complementary outputs. Figures 6c,6d show our implementation style for and/nand and or/nor gates to solve problems with the 2-LAL structure. To our knowledge, this is the first detailed description of 2-LAL implementation of multi-input gates. We now describe the function, the problems and their possible solutions. Like in PFAL (Figure 1c), there are four phases of clock and the stages are pipelined using these. The true and complement inputs of a 2-LAL stage, (which are the outputs of a previous stage) are restored to LOW and HIGH, respectively, during the HOLD phase of the current stage.

In 2-LAL, the restoration is done by the next stage (backward propagation) as will be described shortly. Also, restoration is done only if it is needed, not otherwise that is only if the inputs are not already in their restore states. Figure 6b uses two instances, I1 and I2, of the transmission-gate element of Figure 1a. Like in PFAL (Figure 1c), the clock phase ‘II’ (Phi0 rising) is used to set the inputs INN and IN (previous stage evaluates during Phi0 rising). While in this exercise, we drive these with voltage sources in a way that it complies with the restoration phase, these would be driven by conventional CMOS circuits when integrated into an SoC path. However, for compatibility with 2-LAL first stage operation, these should not be driven during the restoration phase (Phi0N rising) that is tri-stated.

If IN = HIGH (INN = LOW), it would turn ON the instance I1 to pass Phi1 (Phi1N) to OUT (OUTN). Note that while I1 turns ON, Phi1 (evaluation clock phase) is LOW as also the output OUT, thus the drop across the gate is zero. During the subsequent evaluation clock phase (Phi1 rising), the output OUT (OUTN) is driven HIGH (LOW) as Phil (Phi1N) rises (falls). If IN = LOW during the E1 clock phase, then the instance I1 would remain OFF and the outputs remain in their restore state (restored during a previous restore phase—Phi1N rising).

The purpose of I1 is to (a) transfer the input values during the evaluation phase (if inputs were different from the restore state; if not, the outputs are left ‘charged’ in their previous restored state) (b) ‘store’ the evaluated values so as to restore the previous stage’s outputs if needed (i.e. if the stored value is different from restore value) during the restore phase of the previous stage (in the case of primary inputs, Phi0N rising—see Figure 1c) (c) hold the evaluated values on OUT and

**Figure 6** Two-level adiabatic logic (a) Basic transmission-gate element (b) Buffer/inverter (c) And/Nand (d) Or/Nor
OUTN for the next stage to evaluate during the HOLD phase (Phi0N rising, Phi1 falling—see Figure 1c).

For purpose (b) above, the restorations of the previous stage outputs (in this case IN and INN) are done during the HOLD phase of the current stage (in this case, the first stage) through I2 that adiabatically 'passes' Phi0 (Phi0N) to IN (INN) when OUT = HIGH (OUTN = LOW) that is charge restoration happens back to Phi0 from IN (Phi0N charges INN). Thus, the HOLD phase simultaneously enables evaluation of the next stage outputs and restoration of previous stage outputs. Restoration of the inputs IN and INN, turns OFF I2 of Figure 6b (if the inputs IN and INN had changed from their restore state in the first place), thereby allowing the restoration of its outputs, OUT and OUTN, subsequently (Phi1N rising, Phi1 falling). If the inputs had not changed from their restore state during I1 clock phase, then I1 would be in the OFF stage throughout the four phases and OUT and OUTN would also be in the restore state.

This means the next stage would also retain the restore states at its outputs. Note that these restore states are retained ('latched') as charged states ('floating' nodes). Such charged nodes (not driven) would be a concern for serious consideration of SoC and products. The power savings for the 2-LAL buffer/inverter is close to our expectation of 20X (Table 3).

One problem with 2-LAL is backpropagation during restore. A terminating circuit is needed at the end of the pipeline. Figure 7a shows a chain of 2-LAL buffer/inverter cells. The last stage is unterminated. Figure 7b shows the non-adiabatic behaviour of its outputs OUT8/OUTN8. This problem arises because these outputs are never reset (there is no next stage)—its inputs can be corrupted by virtue of instance I2 of Figure 6b being ON when it must be OFF (OUT8/OUTN8, if reset correctly, would have kept I2 OFF and only turned it ON during the evaluation phase based on the input value). The corrupted inputs (outputs of the previous stage, viz. OUT7/OUT7N) can, by the same reasoning, corrupt the outputs of the stage before that, viz. OUT6/OUT6N. OUT5/OUT5N also shows a tendency to switch at the wrong instant (non-adiabatic loss), although it does not make the full transition. The distortion decreases as it propagates backwards.

One fix would be to add the basic Tx gate element of Figure 6a controlled by R phase clock of the last stage (in this case, Phi0N rising: Phi0N/Phi0 to P/PN of Tx gate element, Phi0/Phi0N to A/AN of Tx gate element and connect X/XN of Tx gate element to OUT8/OUTN8). Although this solves the back-propagation, it will cause some non-adiabatic loss in the last stage because the gate inputs to the Tx gate element (Phi0N/Phi0) are changing at the same time as the source terminals (Phi0/Phi0N).

We now turn our attention to the And/Nand gate (Figure 6c) and describe a more severe problem. The input pairs INA/INAN and INB/INBN control two evaluation paths in series (AND: if both INA and INB are HIGH, OUT evaluates to HIGH). The clock-phase description and conditional output evaluation remain the same as the buffer/inverter.

However, restoration of the inputs cannot be done by one path from the reset clocks. If Phi1 rising is E1 phase, Phi0N rising is H1 phase as well as the restore/reset phase for the inputs. The reset clocks Phi0/Phi0N falling/rising reset true and complement inputs through instances I2A and I2B, respectively. Also, the controls for the reset path instances I2A and I2B cannot be OUT/OUTN, which is a combined (series units I1A, I1B) result of both pairs of inputs. We need latched versions of input pairs (like those generated using the forward path of the buffer/inverter of Figure 6b) to control the restore operation for the input signal pairs. These are INAlat/INANlat, INBlat/INBNlat in Figure 6c. The generation of these signals will be discussed shortly.

The Or/Nor gate of Figure 6d can be understood along similar lines. The only difference is the forward path where instances I1A and I1B are in parallel instead of in series. The restore path and its controls are the same as for And/Nand and will be discussed next. The circuits for generating the latched version of an input port (A or B) is shown in Figure 8. The circuits are identical for both ports, so a single circuit is shown with the signal names corresponding to the A and B ports labelled using a wildcard ‘y’ in the name—replace ‘y’ with ‘A’ (Circuit 1 for restoring INA/INAN) or ‘B’ (Circuit 2 for restoring INB/INBN). It is immediately seen that these supporting circuits add many transistors and therefore area to the And/Nand and Or/Nor gates. However, they are essential for the correct operation of these 2-LAL gates.

The evaluation of OUT/OUTN for both Or/Nor and And/Nand was described earlier. We now discuss the supporting reset signals and their generation. These are identical for both gate types and hence we will discuss the operation of

---

**TABLE 3**  
Power of positive-feedback adiabatic, two-level adiabatic, and clocked adiabatic logic versus conventional complementary metal–oxide–semiconductor

| Topology       | Simulation conditions                                                                 | Power, nW (normalised) |
|----------------|---------------------------------------------------------------------------------------|------------------------|
| Conventional   | Load: same gate but with inputs shorted for Nand, Nor; Interconnect = 2F TT, 1.2 G, 125C transitions considered | 1                      |
| Adiabatic PFAL | for power: inverter (4 cycles): rise, fall, rise, fall; Nand/Nor (4 cycles): 10, 00, 11, 01 | 0.0093 0.3434 0.3434   |
| Adiabatic 2-LAL|                                                                                      | 0.056 2.534 3.602      |
| Adiabatic CAL  |                                                                                      | 0.534 0.366 0.366      |

Abbreviations: 2-LAL, two-level adiabatic logic; CAL, clocked adiabatic logic; CMOS, complementary metal–oxide–semiconductor; PFAL, positive-feedback adiabatic logic.

*Phi clock power, when considered, will make these numbers greater than 1.
the first stage of the mixed-logic chain (same as Figure 4a except that the gates are replaced by 2-LAL versions, and signal names do not have the suffix ‘pfal’) that is the Or/Nor gate. The timing waveforms for the Or/Nor operation are shown in Figure 9. The internal signals referred to in Figure 9 belong to the first stage. The internal signals of the Or/Nor (indicated as I15) correspond to the signal names shown in Figure 8a,b. The logic waveforms of operation for the rest of the mixed-logic chain are like those discussed for the PFAL mixed-logic chain except for the internal signals of the gates as discussed below:

First, it is important to note that all the nodes/signals of Figure 8a are targeted for adiabatic function, so they must evaluate and reset at specific phases such that they can recycle at the 100 MHz toggle frequency of the clocks. This is needed to repeat their functions every cycle. The INylat and INyNlat signals are used to control the reset of the inputs. Once that is done, they must be restored to their reset state in keeping with the 2-LAL function as described for Figure 6b. This restoration is done by RSTy and RSTyN_2-LAL signals. These signals are generated using a PFAL stage as shown in Figure 8b. However, the outputs of a PFAL stage are not always complementary. Because these signals are used for a transmission gate, they must always be complementary at all times. Therefore, RSTyN_2-LAL is used instead of RSTyN.

In Figure 8b, RSTyN_2-LAL is generated by the circuit to the right of the PFAL stage (transistors MN18-21, MP7-13) that generates the RSTy signal. The RSTyN_2-LAL is made to track Phi0 while going up and down through the pull-up and pull-down legs. For the time intervals when it can float, transistors MP11-13 drive it, thus preventing a floating node. Unfortunately, this circuit behaves non-adiabatically during its reset, hence resulting in a power loss. This also disrupts the behaviour of INylat/INyNlat. These non-adiabatic behaviours are indicated by ‘*’ in Figure 9. The cause-and-effect lines in Figure 9 show the tracing of the time and value dependencies. The solid circles indicate evaluation at the rising edge of a clock, while the solid squares indicate the reset at the falling edge of a clock. Figure 9 shows the generation of INAlat/INANlat (and INBlat/INBNlat) based on the INA/INAN (INB/INBN) inputs evaluated at the same time (rising Phi1) as the outputs OUTor1/OUTnor1. These are used to control the restore of the inputs of the stage at H1 phase and therefore must be stable at that time. Instances I1 and I2 in Figure 8a enable the conditional evaluation of INylat and INyNlat (where y = A or B for INAlat/INANlat or INBlat/INBNlat) when input INy is HIGH and INyN is LOW and when RSTy and RSTyN_2-LAL are LOW and HIGH, respectively.

The restore path for INylat/INyNlat is through instances I3, I4 and I5—these devices keep the path to the INylat/
In NyNlat signals isolated from the clock until the reset phase. Instance 16 prevents the nodes between I1 and I2 from floating away during the reset phase. The signals RSTy and RSTyN are generated from the PFAL circuit of Figure 8b. Note that the PFAL circuit here does not have the incomplete restore level issue of Figure 4. This is achieved through transistors MN18, MN19, MN4, MN5, MP3 and MP4, which help complete the full swing. The RSTy/RSTyN_2-LAL, along with the inputs of the stage, control the restore of the inputs through instances I3, I4 and I5 of Figure 8a. MN7 isolates INyNlat from RSTyN, thus keeping RSTy/RSTyN stable during INyNlat/INyNlat restore. RST_RSTy/RST_RSTyN are generated by another full-swing PFAL circuit shown in Figure 8b. These are used to reset the RSTy/RSTyN/RSTyN_2-LAL signals and must be stable during that reset phase. Note that each of these PFAL circuits is staggered in its evaluation so as to assert its outputs at the right time to aid the reset of its respective serviced signal. The circuits need to be reset before the next cycle of operations when their action will be required again.

In Figure 9, the INBlat and INBlnlat signals are determined by INB and INBN, respectively. INBlat evaluates HIGH at \( \sim 34 \) ns and again at \( \sim 44 \) ns in response to INB = HIGH. The RSTB* signals are not shown because they behave exactly like RSTA* signals. The logic operation of the *lat signals and the OUT* signals are identical except that the *lat are latched versions with no floating/charged condition. The evaluated latched versions, *lat, thus allow the avoidance of floating nodes by passing their logic function (as shown in Figure 1c for Or/Nor) to the already evaluated OUT* nodes during (using the Hold clock phase, e.g., 'Phi0N rising/Phi0 falling' for the first stage of a chain, as in Figure 7a) the Hold phase when OUT* would otherwise be left to remain charged/float until driven during the subsequent reset phase. This is an adiabatic keeper function.

Simple gates such as Or/Nor and And/Nand in the 2-LAL implementations have some non-adiabatic issues and are more complex than we can afford to use in SoC. In a multimillion-gate SoC, power goes up dramatically, as does the area.

### 3 | COMPARISON AND VALUE FOR SYSTEM-ON-CHIP

Table 3 shows a power comparison for the mixed-gate logic chain context for the three styles against conventional CMOS. The simple PFAL adiabatic buffer/inverter evaluation (Table 2 in the previous section) showed an 8X savings with respect to conventional CMOS at 125°C. However, as discussed, this already reduced to 3X when the voltage changed from 1.26 to 1 V and the temperature decreased from 125°C to 25°C (Figure 2a). The PFAL is the better of the three styles, but when the overhead of clock generation power and interface
problems with conventional CMOS levels are accounted for, it does not seem attractive. The complexity of the 2-LAL Nand/Not discussed earlier is borne out in their excessive power numbers. The power numbers for CAL do not include the clocking power of the Phi clock, which when added will result in power numbers closer to those of conventional CMOS. The body and well currents are less than 1% of the total power in all implementations. The effect of PFET body to Nwell capacitance is accounted for in simulations and therefore included in the power numbers. None of the adiabatic styles are good enough for serious consideration for SoCs.

As mentioned in Section 1, the authors feel that it is more useful to delineate a future research direction for efficient solutions than to present precise layout-based quantification/silicon-based results or derive data for larger block implementations. Moreover, practical issues such as handling fanouts to physically distant (wire delay) instances while adhering to quadrature clock phase functionality, pipelined operation of combinational stages, margin for mismatches, clock variation etc. must be solved. In addition, the high transistor counts in the modified implementations shown, although effective for power, clearly indicate an area of concern. For example, the two-input Nand gate takes 2 N/2P transistors, while the implementations of PFAL, CAL, and 2-LAL are implemented with 6 N/2P, 8 N/2P and 41 N/24P transistors, respectively. The power analyses have been presented on logic chains with a mix of gates and with arbitrary connections as in an SoC as opposed to a regular structure like an adder. The latter is a structured staged/pipelined circuit that is more favourable to adiabatic styles and not truly representative of the millions of gates used in SoC. Structured circuits are important to evaluate, too, but our goal was to understand the suitability to randomly interconnected combinational logic.

A modest performance of 100 MHz, a 20X power savings with respect to conventional CMOS so as to net a savings of at least 10X after accounting for clocking power, less than a 5% die size area increase, and compatibility with existing SoC tools and methodologies (including library views, synthesis, place/route and timing closure) with a recognition of multiphase clocking would be the broad goals for serious adoption of adiabatic circuits.

It must be acknowledged that there are some lower-speed applications with high premiums on low power and less constraints on area, such as medical applications. These could even be custom designed due to low gate counts. Clocking schemes using external components are also possible. However, care must be taken to address the fundamental circuit issues described in this paper.

4 | CONCLUSION

We evaluated three adiabatic circuit styles for power savings in an SoC context, expecting to save 20X in power. Unlike other published material, where simple or regular structures have been used to demonstrate power savings, we evaluated random-logic chains including a mix of buffers/inverters, hands, nors, and multiplexers at 100 MHz and found circuit-level issues that make it impractical to realise the targeted power savings. Interfacing conventional full-swing CMOS to adiabatic circuits was shown to be a first-order problem for the first time. We showed a modified PFAL circuit without the incomplete restore level problem (Figure 8b). The 2-LAL circuits developed do not have floating or charged nodes. The analysis of 2-LAL multi-input gates and associated problems and solutions were described here for the first time. The solutioning for the 2-LAL consisted of (a) generating data-dependent synchronous reset signals using cascaded PFAL stages; (b) PFAL stages modified to reduce non-adiabatic losses; (c) using these reset signals along with existing clocks to latch the 2-LAL first-stage outputs; and (d) using these latched signals to ensure correct operation of multi-input 2-LAL gates. While our work was in 40 nm CMOS technology, the principles explained herein are applicable to implementations in any other technology as well—the results may vary due to changes in process and voltage. We realise that radical changes are required to logic styles for them to become competitive in terms of power, performance and area. Future work will hopefully address these weaknesses.

ORCID

Krishnan S. Rengarajan https://orcid.org/0000-0001-5967-3005

REFERENCES

1. Blotti, A., Saletti, R.: Ultralow-power adiabatic circuit semi-custom design. IEEE Trans. Very Large Scale Integr. Syst. 12(11), 1248–1253 (2004)
2. Campos-Aguillón, C. O., et al.: A Mini-MIPS microprocessor for adiabatic computing. In: Proceedings of the 2016 IEEE International Conference on Rebooting Computing (ICRC), pp. 1–7. IEEE, San Diego (2016). https://doi.org/10.1109/ICRC.2016.7738678
3. Erik, P.D., et al.: A path towards ultra-low-energy computing. In: Proceedings of the 2016 IEEE International Conference on Rebooting Computing (ICRC). IEEE, San Diego (2016). https://doi.org/10.1109/ICRC.2016.7738677
4. Younis, S.G.: Asymptotically zero energy computing using split-level charge recovery logic. Dissertation, Massachusetts Inst of Tech Cambridge Artificial Intelligence Lab, Massachusetts (1994)
5. Vertuli, A., Pascoli, S.D., Reynen, L.M.: Positive feedback in adiabatic logic. Electron. Lett. 32(20), 1867–1869 (1996)
6. Frank, P.M.: Towards a more general model of reversible logic hardware. Superconducting Electronics Approaching the Landauer Limit and Reversibility (SEALeR) Workshop, Annapolis (2012). http://www.english.uidaho.edu/~mpf/MPF-SEALeR-talk.pdf
7. Maksimovic, D., et al.: Clocked CMOS adiabatic logic with integrated single-phase power-clock supply. IEEE Trans. Very Large Scale Integr. Syst. 8(4), 460–463 (2000)
8. Wang, B., Mazumder, P.: On optimality of adiabatic switching in MOS energy-recovery circuit. In: Proceedings of the 2004 International Symposium on Low Power Electronics and Design (IEEE Cat. No.04TH8758), pp. 236–239. IEEE, Newport Beach (2004). https://doi.org/10.1109/LPLE.2004.241029
9. Celis-Cordova, R., et al.: Design of a 16-Bit Adiabatic Microprocessor. In: Proceedings of the 2019 IEEE International Conference on Rebooting Computing (ICRC), 1-4 IEEE, San Mateo 2019. https://doi.org/10.1109/ICRC.2019.8914699
10. Kattir, M., Ejlali, A., Moradi, A.: Improving the energy efficiency of reversible logic circuits by the combined use of adiabatic styles. Integration. 44(1), 12–21 (2011)
11. Lim, J., et al.: nMOS reversible energy recovery logic for ultra-low-energy applications. IEEE J. Solid State Circ. 35(6), 865–875 (2000)
12. Lim, J., et al.: Reversible energy recovery logic circuit without non-adiabatic energy loss. Electron. Lett. 34(4), 344–345 (1998)
13. Frank, P.M.: Common mistakes in adiabatic logic design and how to avoid them. Embedded Systems and Applications (2003). https://www.semanticscholar.org
14. Moon, Y., et al.: An efficient charge recovery logic circuit. IEEE J. Solid State Circ. 31(4), 514–522 (1996)
15. Avital, M., et al: DPA-secured quasi-adiabatic logic (SQAL) for low-power passive RFID Tags Employing S-Boxes. IEEE TCAS-I (2015)
16. Kumar, D.S., et al.: Energy-Efficient and Secure S-Box Circuit Using Symmetric Pass Gate Adiabatic Logic. In: Proceedings of 2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), pp. 308–313. IEEE, Pittsburgh (2016). https://doi.org/10.1109/ISVLSI.2016.45
17. Lim, J., et al.: A 16-bit carry-lookahead adder using reversible energy recovery logic for ultra-low-energy systems. IEEE J. Solid State Circ. 34(6), 898–903 (1999)
18. Oklobdzija, G.V., et al.: Pass-transistor adiabatic logic using single-power-clock supply. IEEE TCAS-II (1997)
19. Chanda, M., et al: Comparative Analysis of Adiabatic Compressor Circuits for Ultra-low Power DSP Application. In: Proceedings of 2010 International Conference on Advances in Recent Technologies in Communication and Computing, pp. 355–359. IEEE, Kottayam (2010). https://doi.org/10.1109/ARTCom.2010.78
20. Chanda, M., et al.: Implementation of subthreshold adiabatic logic for ultralow-power application. IEEE Trans. Very Large Scale Integr. Syst. 23(12), 2782–2790 (2015)
21. Arsalan, M., et al.: Comparative analysis of adiabatic logic styles. In: Proceedings of 8th International Multitopic Conference, 2004, pp. 663–668 IEEE, Lahore (2004). https://doi.org/10.1109/INMIC.2004.1492971
22. Ye, Y., Roy, K.: QSERL: quasi-static energy recovery logic. IEEE J. Solid State Circ. 36(2), 239–248 (2001)
23. Takahashi, Y., et al.: 2PCDAL: two-phase clocking dual-rail adiabatic logic. In: Proceedings of the 2012 IEEE Asia Pacific Conference on Circuits and Systems, pp. 124–127. IEEE, Kaohsiung (2012). https://doi.org/10.1109/APCAS.2012.6418987
24. Hu, X., Guthaus, R.M.: Distributed LC resonant clock grid synthesis. IEEE Trans. Circuits Syst. I Regul. Pap. 59(11), 2749–2760 (2012)

How to cite this article: Rengarajan KS, Mondal S, Kapre R. Challenges to adopting adiabatic circuits for systems-on-a-chip. IET Circuits Devices Syst. 2021;1–13. https://doi.org/10.1049/eds2.12053