

# Resonant System Design with Coarse Grained Pipelines

Visvesh S. Sathe, Marios C. Papaefthymiou  
Department of EECS, University of Michigan  
Ann Arbor, USA  
{vssathe,marios}@eecs.umich.edu

## Abstract

*In this report, we present an efficient approach to resonant system design. Our approach involves the use of resonant clocks to drive level sensitive latches in pipelined datapaths. Through judicious design of these timing elements, the energy efficiency of resonant clocking can be obtained without performance penalties, while maintaining robust, race-free operation. Since our approach involves driving only the timing elements with resonant clocks and places no restrictions on the type of computational logic, the method can be used with existing static CMOS design flows. We describe our technique for two, three and four phase clock systems and present clock generation mechanisms. We also introduce the level-sensitive timing elements to be used with these clocks and discuss how they are introduced into a datapath.*

## 1 Introduction

Power minimization continues to be a critical issue in many VLSI designs today. Excessive power dissipation can result in increasing packaging and cooling costs and reduce the operating lifetime of the system for a given amount of energy storage.

It is well known that a significant amount of energy dissipation in synchronous systems is dissipated in the clock tree and nodes of a design. High performance designs tend to be heavily pipelined and therefore present a very large switching load in the form of the clock capacitance. Furthermore, many energy critical applications utilize the technique of pipelining and subsequent voltage scaling to achieve energy reduction. For both these systems, a significant portion of the total energy dissipation occurs in the clock. As such, any methodology which reduces the energy dissipation of the clock in a given design without compromising on its performance significantly impacts overall system efficiency.

Resonant clocking schemes have been previously proposed with the objective of minimizing clock dissipation. While predominantly used in energy recovery systems in the form of power clocks driving adiabatic logic, resonant clocks have the potential to be utilized in conventional systems as well. To significantly reduce the energy dissipation in the clock using resonant clocking schemes, it is necessary to resonate clock capacitance all the way down to the timing elements. However, a major drawback of resonant clocks is the sinusoidal nature of these clock waveforms. The slew provided by these clocks is unacceptable to most flip-flop designs. While buffers can be inserted to provide the required slew to these flip-flops, there are significant drawbacks to this approach. The use of buffers limits the overall clock capacitance that can be resonated since all the down-stream capacitance after the buffer, including the flip-flop load cannot be resonated. Furthermore, the crowbar current involved in converting a sinusoidal resonant waveform to the desired slew can be significant.

In this paper, we discuss the design of an entire class of datapaths clocked by resonant waveforms. We refer to such systems as resonant systems. Previous work in the design of such resonant systems has focused on the design of flip-flops that would work with a sinusoidal waveform [1]. However, as mentioned, such methods fundamentally derive a latching instant (a hard edge) based on the voltage of the resonant clock in order to capture data. Such systems are vulnerable to process variation which effectively injects clock skew into the design. Furthermore, obtaining an edge from the resonant waveform for clocking the flip-flops precludes resonating a large percentage of the overall clock power. The scheme proposed by [2] connects the resonant clock to transistor drains which enables more capacitance

in the flop to be resonated. However, this approach degrades the efficiency of the circuit and is susceptible to hold-time violations. The proposed work proposes the use of timing elements operating on soft edges (latches) with resonant clocks. Such a scheme has minimal dependence on the slew of the clock waveform and can be designed to be free of hold-time violations over significant amounts of skew. However, the timing properties of the latches are affected by the voltage of the clock at the time of data transmission. By designing the system to ensure that data arrives at latches at or before the time the latch is most strongly transparent, the performance degradation incurred in the use of sinusoidal clock waveforms is reduced.

In addition to the latch designs, we present novel self-resonating clock generators for the purpose of generating two, three and four phase waveforms. By resonating the switches that provide the negative conductance to the oscillators, higher energy efficiency can be achieved.

The remainder of this paper is organized as follows. In Section 2, we discuss the design of skew tolerant latch-based design in the context of resonant systems and show how multiple clock phases and clock generation methodologies enable the design of efficient, resonant systems. We also discuss resonant clocking, some of the latches that we have designed for use in resonant systems and the accompanying clock waveforms that are required to make them work. The timing properties of these latches are also discussed in this section. Finally, in section 3, we discuss clock generator designs that enable the efficient generation of resonant clock waveforms.

## 2 Latch Design with Resonant Clocks

In this section, we outline the necessary conditions that clock phases of latch based designs need to meet in order to remain skew-tolerant and show how these conditions can be met using traditional sinusoidal resonant clock waveforms. We also give specific latch designs that are well-suited for resonant clocks. Conventional skew-tolerant design incorporates the use of two, non-overlapping clock phases which are used to control the timing operation of latches to avoid race conditions. Generating two non-overlapping clock phases requires the generation of two waveforms with less than 50% duty cycle. Furthermore the clock waveforms are more or less trapezoidal in nature, resulting in a region of relatively constant D-Q delay in the latch. It is easily seen that no two sinusoidal waveforms can be time-shifted with respect to each other to become non-overlapped. Furthermore, any techniques that try to change the shape of the naturally occurring sinusoidal waveform limit the efficiency of a resonant system.

To implement pipelined resonant systems efficiently therefore, a method of generating or inferring non-overlapped regions of latch transparency is necessary. In addition, although latch-based design does not require sharp clock transitions for performance or energy efficiency, data transfer across pipelines must take place while ensuring that data on the critical path arrives at the time when the clock is around its peak. More generally, the system is designed that data arrives at the input of the latch no later than the “peak” of its conducting phase. This prevents any performance degradation in the system. Thus, unlike most conventional latch-based designs where data is intended to arrive at a latch just after it becomes transparent to provide for the possibility of aggressive time borrowing, resonant pipelines are designed so that data inputs arrive at the latch not at the onset of latch transparency, but at the peak of the clock cycle, corresponding to the center of the high-time of conventional clocks. While the sinusoidal nature of the clock waveform limits the extent of time borrowing that can be achieved, it removes the timing penalty associated with the D-Q delay in the latch as a result of being clocked by a sinusoidal waveform. The cycle time,  $T_{cycle}$  of such a resonant datapath, is set as follows:

$$T_{cycle} = T_{DQ}(T_{skew}) + T_D, \quad (1)$$

where  $T_{DQ}$  is the DQ delay of the latch when data arrives while the latch features the least DQ delay (note that  $T_{DQ}$  is a function of the possible clock skew), and  $T_D$  is the critical path delay in the logic. Signals arriving at latches before the clocking signal reaches its peak will incur a higher D-Q delay but they are guaranteed to arrive at the output of the latch before the critical data arrives at the output and therefore remain non-critical.

One way to generate non-overlapping resonant waveforms is the use of the blip-generator. The modified blip generator shown in Figure 1 generates almost-non-overlapping waveforms. The use of the conventionally driven switches can be used to control the extent of the overlap between the waveforms to guard against possible hold time violations in the presence of clock skew. Furthermore, since both  $\phi$  and  $\bar{\phi}$  routes are present across the design the cross-coupled switches can be deployed throughout the design amidst the cells so as to reduce local skew between the two clocks.

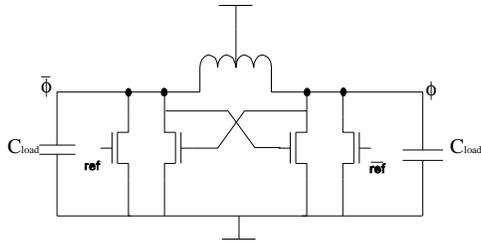


Figure 1: Modified Blip Generator Schematic

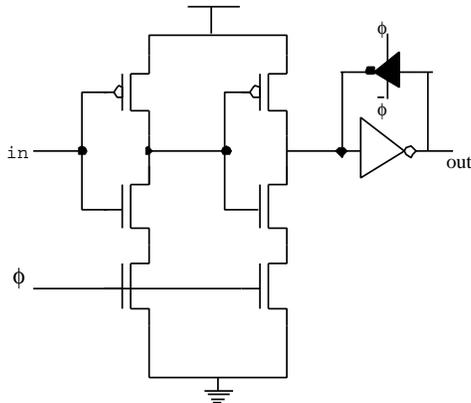


Figure 2: Two-phase Resonant Latch Schematic

With the non-overlapping clock waveforms readily available, the design of latches operating with these waveforms is straightforward. An example of such a latch is displayed in Figure 2

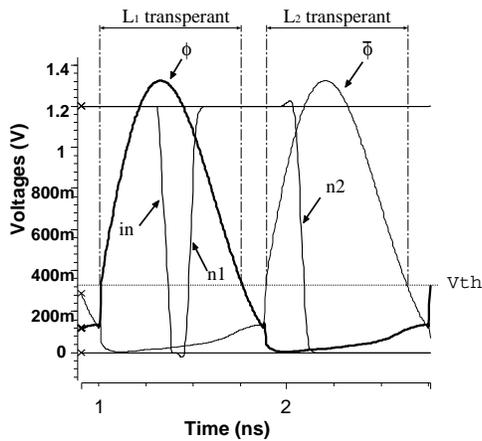


Figure 3: Blip Latch Simulation Waveforms

Figure 3 shows waveforms obtained from spice simulations of a data input propagating through two consecutive two-phase resonant latches without any logic between them. Note how the non-overlapping clocks ensure that short paths do not result in a race. It can be shown that a hold time violation does not occur in this structure over substantial amounts of clock skew.

While the blip generator implementation is the most straightforward, the blip generator is not an efficient clock generation mechanism due to the substantial losses in the inductor and the cross-coupled switches during the current

build-up in the inductor. In the rest of the section we show how it is possible to derive the required non-overlapping transparent phases in a clock cycle.

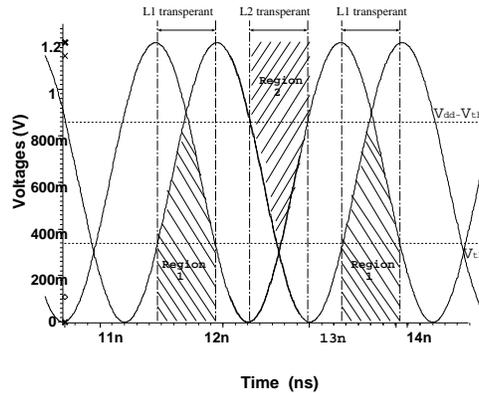


Figure 4: Sinusoidal waveforms used to define non-overlapping transparent phases

Figure 4 shows two clocks,  $\phi$  and  $\phi'$  with some arbitrary phase shift. While the clock waveforms overlap substantially, consider the shaded regions shown in the figure. Region 1 is the result of the intersection of the two waveforms while the voltage of both waveforms is greater than the threshold voltage  $V_{th}$ . Region 2 is the result of the intersection of the two waveforms while they are below  $V_{dd} - V_{th}$ . Clearly, if the latches were designed to be transparent in the two regions, then the two clocks could be used to clock such latches.

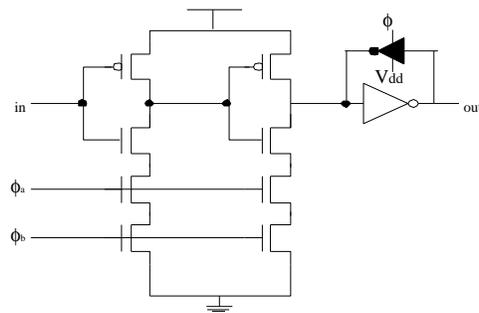


Figure 5: N-type latch used for two overlapping sine waveforms

Figure 5 shows two latches which can be clocked by waveforms shown in Figure 4. The n-type latch remains transparent throughout Region 1, while the p-type latch remains transparent throughout Region 2. Thus a pipelined design using these latches uses alternating types of n and p-type latches.

Figure 6 gives a spice waveform showing that two back-to-back two-phase resonant latches can be connected together without a race. (It has been experimentally determined that a hold time violation does not occur in this structure over large skew)

Figure 7 shows a latch which uses two clocks  $120^\circ$  out of phase. The latch works on the same principle described previously of inferring non-overlapping transparent windows. The proposed three phase np latch operates with a nearly ideal  $60^\circ$  overlap. This is done by using an n-p latch structure such that the transparent window is defined by the amplitude of the first clock and the difference between the supply and amplitude of the second clock.

Figure 8 shows how two back-to-back np latches are connected together without a race. (It has been experimentally determined that a hold time violation does not occur in this structure over large skew)

Figure 4 also suggests a trade-off in choosing the extent of the phase shift  $\phi$ . Notwithstanding the fact that certain clock phases can be generated more efficiently as compared to others, the choice of the phase shift between the two clocks has opposite effects on the D-Q delay of the latches and the skew tolerance of the design. While reducing the

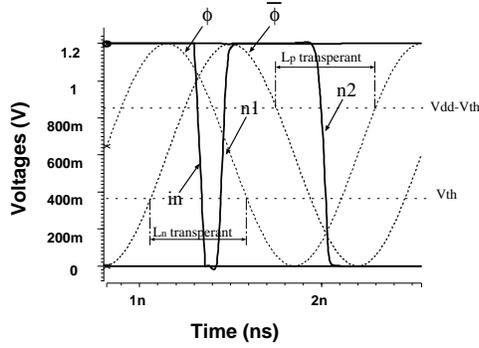


Figure 6: Spice waveform showing data passing through two resonant latches with  $\phi_a - \phi_b = 90^\circ$

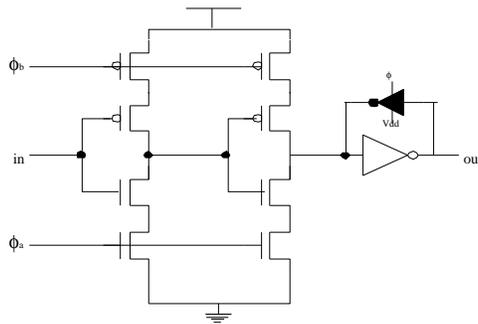


Figure 7: NP-latch for  $120^\circ$  out-of-phase clock waveforms

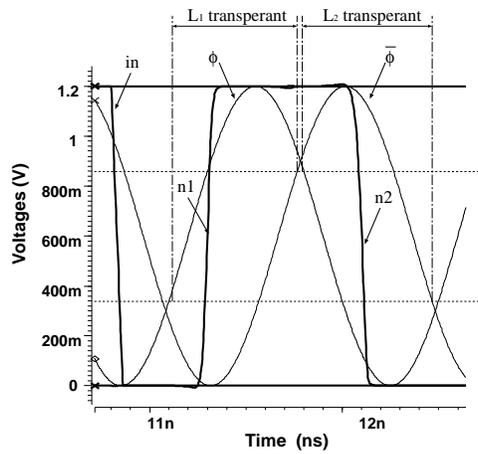


Figure 8: Spice waveform showing data passing through two resonant latches with  $\phi_a - \phi_b = 120^\circ$

phase shift increases the amplitude of the clock intersection, decreasing D-Q delay, it increases the overlap between the two conducting regions. In conventional latch-based designs driven by clocks with high slew, overlaps between alternating clock phases resulting in a significant reduction in skew tolerance. In systems clocked with sinusoidal waveforms, however, significant overlap between the conducting regions is possible while retaining tolerance to clock skew.

The reason behind this retained robustness to clock skew even with overlapping conducting clock phases is simple. Equation 2 can be used to determine the amount of allowable overlap between clock phases.

$$V_{skew} = T_D + T_{DQ}(\tau_a) - T_{hold} + T_{not}, \quad (2)$$

where  $T_D$  is the minimum delay in the logic path,  $T_{DQ}$  is the D-Q delay of the latch and is a function of the data arrival time at the latch,  $T_{hold}$  is the hold time of the latch and  $T_{not}$  is the time difference between the alternating conducting phases of the system. With sinusoidal clock inputs such as shown in Figure 8, minor overlaps in the conducting regions of serially connected latches occurs with a low clock overdrive, due to the sinusoidal waveform of the clocks. Data arriving at the first latch in the overlap region would exhibit a much larger D-Q delay than when arriving at the peak of the conducting phase. This observation, along with Equation 2 explains

In the next section, we discuss some of the mechanisms with which these resonant clocks are generated. In particular, we propose efficient ways to generate clock waveforms  $90^\circ$  and  $120^\circ$  out of phase.

### 3 Resonant Clock Generators

In this section, we discuss some efficient self-resonating clock generators capable of generating the clock phases required in skew-tolerant resonant latch design. In particular, we will discuss clock generators of  $90^\circ$  and  $120^\circ$  phase shift clocks that are totally free-running and do not require any drive stages to re-inject energy into the system. These self-resonating clock oscillators afford higher efficiency as compared to the traditional power clock generator topology [3] by resonating the gate capacitance of the the switches used to periodically inject energy into the system. However, driven oscillators have the advantage that energy is provided into the system with minimal switching loss. In order to design efficient self-resonating clock generators therefore, it must be ensured that switching losses are kept to a minimum. Clearly, the phase difference between the clocks plays an important part in this aspect. It will be observed that the switching losses are minimal for the three phase clock generators, while being higher for the four-phase clock generator and substantially higher for the blip generator.

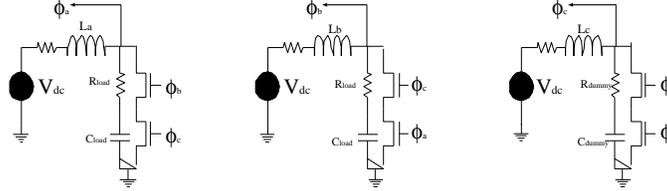


Figure 9: Simple three-phase clock generator

Figure 9 shows one technique to generate a three phase clock. The two capacitive loads  $C_{load}$  are the loads of the clock network as distributed throughout the chip. The third load,  $C_{dummy}$  is connected to the third phase  $\phi_c$ , which is not propagated to the design and is implemented as a MOS capacitor. The three phases, namely  $\phi_a$ ,  $\phi_b$  and  $\phi_c$ , reach a stable oscillation  $120^\circ$  out of phase. To explain the operation of the the above clock generator, we first observe from Figure 8 that in a three-phase system, the phase difference between the clocks can be used to provide the replenishing current to each section of the clock generator. By using a series connection of switches driven by the other two clock phases, each clock phase obtains the required current injection into the inductor while the output is low and switching losses are at a minimum. Furthermore, since the capacitive load on the three phases is equal, the currents flowing through the inductors  $L_a$ ,  $L_b$  and  $L_c$  are  $120^\circ$  out of phase. As a result, the current flow through the dc supply,  $V_{dc}$  is zero. The three inductors can therefore be connected in a star configuration and the dc supply can be removed without changing the behavior of the circuit as long as the dc-solution of the star-junction is maintained.

Figure 10 shows a topology that provides the same functionality as the clock generator proposed in Figure 9. However, while the standard three-phase clock generator shown in Figure 10 would serve to function as an efficient three-phase oscillator, it is not well suited to clock generation because the coupling capacitance between actual distributed clock networks is a significant contributor to the overall capacitive load seen by the clock generator. The capacitive coupling between the clock phases  $\phi_a$  and  $\phi_b$  destroys the symmetry in the clock generator configuration and can lead to improper operation. In order for the three-phase scheme shown above to work satisfactorily, careful

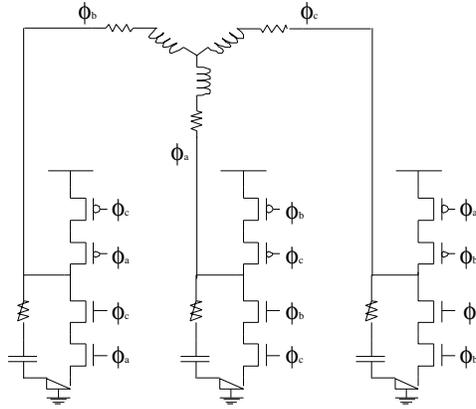


Figure 10: Three-phase clock generator in star configuration

clock design is required to ensure that the coupling capacitance between all three phases is balanced. This balancing can often be undesirable for clock design.

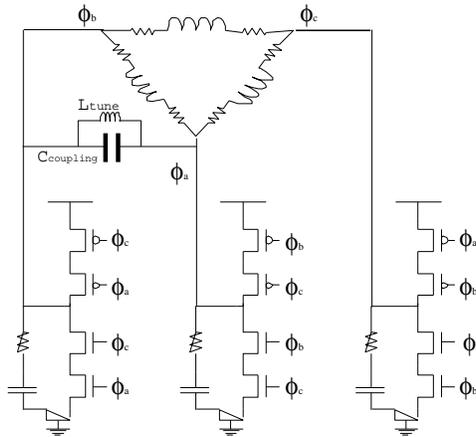


Figure 11: Three-phase clock generator in delta configuration

To solve this problem, consider the equivalent delta configuration of the inductors shown in Figure 11. The coupling capacitance between  $\phi_a$  and  $\phi_b$  can be tuned out at the required resonant frequency with the addition of  $L_{tune}$ . This inductor appears in parallel with  $L_C$  and therefore only one inductor, the parallel combination of the two inductors, needs to be implemented.

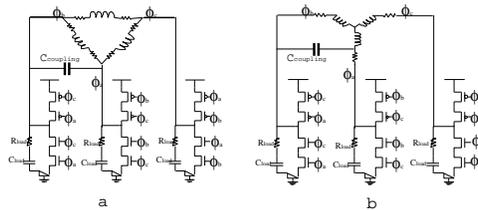


Figure 12: Coupling Compensated Clock generator in (a) Star and (b) Delta mode

The coupling capacitance compensated clock generator can be implemented as shown in either Figure 12a or

Figure 12b. The star configuration shown in Figure 12b is obtained by simply transforming the delta configuration impedances to star configuration impedances.

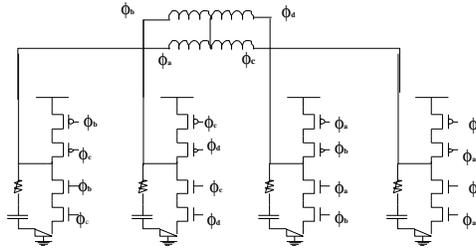


Figure 13: Four-phase clock generator in delta configuration

The design of a four-phase resonant clock generator is based on a similar principle. Figure 13 shows the clock generator topology for a four phase clock topology. Here, instead of four inductors, one for each phase, two center-tapped inductors are used. The center taps are then connected together and the series connected pullup and pull-down devices provide the negative conductance needed for sustained oscillations. It is interesting to note that a four-phase oscillator proposed in [4] has a somewhat similar topology. Other four phase LC oscillators have also been proposed [5]. While the proposed topology will have a significantly higher distortion as compared to traditional oscillator designs, the latter were not designed for distributed clock networks and trade-off energy efficiency for other properties such as phase noise and distortion. The intended use for this topology will be in the design of resonant latch systems of the kind described in Figure 5. An additional use of the four-phase system is in the efficient generation of two-phase resonant clocks. The four phases can be routed as  $\phi$  and  $\bar{\phi}$  in two separate clock domains.

## 4 Conclusion

We have outlined a methodology for designing skew-tolerant resonant latch-based systems. These systems have the potential to drastically reduce the energy dissipated in the clock network by resonating the clock distribution network up to and including the timing elements at the leaves of the clock tree. Measurements from spice simulations show that these timing elements do not present a performance overhead to the system. The use of resonant latch-based systems will also have a significant impact on decisions of optimal pipelining and voltage scaling arising from pipeline stages with minimal energy overhead and skew invariance in resonant clock distributions from voltage scaling.

## References

- [1] M. Cooke, H. Mahmoodi-Meimand, and K. Roy, "Energy recovery clocking scheme and flip-flops for ultra low-energy applications," pp. 54–59, 2003.
- [2] C. Ziesler, J. Kim, V. Sathe, and M. Papaefthymiou, "A 225 MHz resonant clocked ASIC chip," in *ISLPED*, Aug 2003.
- [3] C. Ziesler, S. Kim, and M. C. Papaefthymiou, "Resonant clock generator for single-phase adiabatic systems," in *ISLPED*, Aug 2001.
- [4] P. Andreani, A. Bonfanti, L. Romano, and C. Samori, "Analysis and design of a 1.8-GHz CMOS LC quadrature VCO," *JSSC*, pp. 1737–1747, Dec 2002.
- [5] R. Rofougaran, A. Rael, M. Rofougaran, and A. Abidi, "A 900 mhz cmos lc-oscillator with quadrature outputs," pp. 392–393, Feb.