A New Low Power High Performance Flip-Flop

Ahmed Sayed and Hussain Al-Asaad
Department of Electrical and Computer Engineering
University of California
Davis, CA, U.S.A.

Abstract*—Low power flip-flops are crucial for the design of low-power digital systems. In this paper we delve into the details of flip-flop design and optimization for low power. We compare the lowest power flip-flops reported in the literature and introduce a new flip-flop that competes with them.*

1 Introduction

As the feature size of CMOS technology process shrinks according to Moore's Law, designers are able to integrate more transistors onto the same die. The more transistors there are the more switching and the more power dissipated in the form of heat or radiation. Heat is one of the most important packaging challenges in this era; it is one of the main drivers of low power design methodologies and practices. Another mover of low power research is the reliability of the integrated circuit. More switching implies higher average current is flowing and therefore the probability of reliability issues occurring rises.

The most important prime mover of low power research and design is our convergence to a mobile society. We are moving from desktops to laptops to handhelds and smaller computing systems. With this profound trend continuing, and without a matching trend in battery life expectancy, the more low power issues will have to be addressed. This entails that low power tools and methodologies have to be developed and adhered to. The current trends will eventually mandate low power design automation on a very large scale to match the trends of power consumption of today's integrated chips.

There are many flip-flops that have been reported in the literature. Some of these flip-flops are quite good at being low power and high performance. In this paper, we compare the best 3 flip-flop circuits as reported in [1] with a new proposed circuit that we developed. The rest of this paper is organized as follows. Section 2 presents background information about flip-flop design and characteristics. Section 3 presents the studied flip-flop circuits with a short description of each flip-flop followed by the introduction of our new flip-flop design. Section 4 presents the simulation and evaluation results of these flip-flops. Finally, Section 5 presents some discussion and conclusions.

2 Background

2.1 Power Consumption in Logic Circuits

The instantaneous power of any circuit is calculated as follows:

$$P(t) = i_{dd}(t)V_{dd} \tag{1}$$

The above equation assumes that the voltage power supply is stable and constant throughout operation. The energy consumed over the time interval T is the integral of the instantaneous power:

$$E = \int_0^T i_{dd}(t)V_{dd}dt \tag{2}$$

The average power used over the interval is just the energy divided by the time:

$$P_{avg} = \frac{E}{T} = \left(\frac{1}{T}\right)\int_0^T i_{dd}(t)V_{dd}dt \tag{3}$$

For CMOS digital circuits, equation (3) can be further expressed in the following equation:

$$P_{avg} = p_t(C_L V V_{dd} f_{clk}) + I_{sc}V_{dd} + I_{leakage}V_{dd} \tag{4}$$

The above equation consists of three terms and hence illustrates that there are three major sources of power consumption in a digital CMOS circuits. The first term represents the switching component of power, where C_L is the effective switched loading capacitance, f_{clk} is the clock frequency and p_t is the probability that a power consuming transition occurs (referred to as the activity factor in other publications). In most cases, the voltage swing V is the same as the supply voltage V_{dd}. However, in some logic design styles such as in pass-transistor logic, the voltage swing on some internal nodes may be slightly less. It is important to point out, that the effect of internal glitching (to be discussed later) should be included as a component of the switching power consumption. The second term is caused by the direct path short circuit current I_{sc}, which arises when both the NMOS and PMOS transistors or networks are simultaneously active or on, conducting current from the supply V_{dd} to ground. Finally, a factor that is growing more and more important as we develop deep submicron technologies, leakage current $I_{leakage}$, which can arise from substrate injection, gate leakage and subthreshold effects. $I_{leakage}$ is primarily determined by the CMOS fabrication process technology and modeled based on its characterization.

We can observe from (4) that power consumption of a circuit depends strongly on its structure and input data statistics. All the nodes of a circuit contribute to the total power consumption of the circuit so (4) should be applied to each and every node at a micro scale. An alternative is that the designer might like to break the power consumption down into internal and external components, which identify the internal inherent power consumption of the circuit and the external effect of the load on the power consumption. The internal power consumption can be broken down into the following:

- Internal power consumption of the flip-flop or latch.
- Local clock power consumption that is consumed in the local clock buffer driving the internals of the latch or flip-flop.
- Local data power consumption that is consumed in the logic stages and transistors of the latch or flip-flop driven by the data input and driving the data output.

This breakdown is not followed in this paper since our goal is to perform system level comparisons of different circuits rather than optimizing certain metrics and power components of the flip-flops being compared.

The dominant term in a well-designed circuit is the switching component, thus the low-power design goal becomes the task of minimizing $p_t(C_L V V_{dd} f_{clk})$, while retaining the required functionality and identifying the cost of such minimizations in terms of area and/or performance.

The peak power consumption could be very useful when trying to find out the worst case scenario for your design or system, for example, the worst case of battery life expectancy of your laptop or cell phone. This is measured as the worst case or maximum instantaneous current drawn from the supply within a specific time period of interest and is expressed as:

$$P_{peak} = max(i_{dd}(t))V_{dd} \tag{5}$$

We chose the peak power consumption to be measured because this is really the parameter to be concerned with during the design phase of a system. The clock and power delivery networks should be capable of withstanding the peak power consumption of the system without failing. Average power is a good metric for the goodness of the circuit and how much power would be used on average, but is dependent on activity and switching probabilities, which in turn are very dependent on the application.

The peak power measurement is quite problematic in general logic circuits, the reason behind this statement is the difficulty of establishing and qualifying the set of input transitions i.e. vectors and relative timings that cause the circuit to consume most power. This is a very tough issue to solve in generic designs or circuits, but not that bad for flip-flop circuits as the number of inputs is limited and the relative timings are direct forward, i.e. within the clock period of operation.

The power-delay product (PDP) can be viewed as the amount of energy expended in each switching event and is thus particularly important in comparing the power consumption of various circuits and design styles. Assuming that the full swing switching component of (4) is dominant, this metric becomes:

$$PDP_{avg} = p_t(C_L V V_{dd} f_{clk})/f_{clk} = p_t(C_L V_{dd}^2) \tag{6}$$

A more performance oriented metric for circuits and design styles would be the energy-delay product. This is considered if performance is of a higher importance and priority than power consumption. This will not be used here since low power is our highest priority.

In this paper, we will refer to PDP as the product of the peak power consumed and the D-to-Clock delay. This will be one of the used comparison metrics among flip-flops.

2.2 Basics of Sequential Elements

Sequential elements are mainly used to store computation result values for future use. At the minimal level of storage, an element should be able to store a logic “1” or “0” value reliably.

Transitions on the inputs of a flip-flop may or may not lead to a state change. When input transitions do not change the state, the internal switching inside the flip-flop consumes some power. On the other hand, when the input transitions do change the state, a bigger amount of power is consumed.

Flip-flops can be classified in several ways: dynamic vs. static, square-wave vs. pulsed, conditional vs. non-conditional, and depending on the logic style used. In this paper we consider different flip-flops with different classifications.

2.3 Flip-Flop Comparison Metrics

There are several basic performance metrics that are used to qualify a flip-flop and compare it to other designs. These metrics are:

- Clock-to-Q delay: Propagation delay from the clock terminal to the output Q terminal. This is assuming that the data input D is set early enough with respect to the effective edge of the clock input signal.
- Setup time: The minimum time needed between the D input signal change and the triggering clock signal edge on the clock input. This metric guarantees that the output will follow the input in worst case conditions of process, voltage and temperature (PVT). This assumes that the clock triggering edge and pulse has enough time to capture the data input change.
- Hold time: The minimum time needed for the D input to stay stable after the occurrence of the triggering edge of the clock signal. This metric guarantees that the output Q stays stable after the triggering edge of the clock signal occurs, under worst PVT conditions. This metric assumes that the D input change happened at least after a minimum delay from the previous D input change, this minimum delay is the setup time of the flip-flop.
- Data-to-Q delay: The sum of setup of data to the D input of flip-flop and the Clock-to-Q delay as defined above.

Library developers often try their best to minimize the setup time requirement of flip-flops and the Clock-to-Q delay since most synchronous designs are targeting the most design performance at hand. Specifically in pipelined designs, where flip-flop “setup time + Clock-to-Q delay” is a main constraint of the maximum clock frequency of operation for a given function, which in turn mandates the number of stages needed to perform the required function and affects in turn the latency and throughput of the whole design.

Hold times are not as critical as setup times and they do not limit the speed of a circuit in flip-flop based designs. On the other hand they are very critical in latch-based designs.

2.4 Regions of Flip-Flop Operation

There are three regions of flip-flop operation, of which only one region is acceptable for a sequential design to function correctly. These regions are:

- Stable region: Where the setup and hold times of a flip-flop are met and the Clock-to-Q delay is not dependent on the D-to-Clock delay. This is the required region of operation.
- Metastable region: As D-to-Clock delay decreases, at a certain point the Clock-to-Q delay starts to rise exponentially and ends in failure. The Clock-to-Q delay is nondeterministic and this might cause intermittent failures and behaviors which are very difficult to debug in real circuits.
- Failure region: Where changes in data are unable to be transferred to the output of the flip-flop.

Figure 1 illustrates the different regions of flip-flop operation. The optimal setup time noted on the graph would be the highest performance D-to-Clock delay to accomplish fastest D-to-output delay. Due to the steep curve to the left of that point not all library

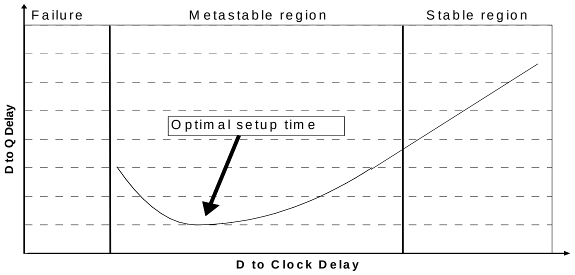

Figure 1 Flip-flop regions of operation.

developers would target this value. Instead, they would prefer adding guard bands to any library cell or design to guarantee stability and reliability.

2.5 Hazards and Glitches

We define a glitch to be any spurious transient output in combinational circuits. There are various phenomena that can cause glitches and the main one is hazards in combinational circuits. If the output signals for a combinational network depend on the internal circuit delays, elements and interconnects, as well as on input signals, the circuit is said to contain a hazard. There might be other causes of hazards in a circuit, for example the relative delays of the asynchronous inputs might exacerbate a hazard scenario which was not supposed to occur. Unequal delay paths in a circuit are a very common cause of hazards in combinational circuits. Interconnect delays are becoming more and more significant with submicron technologies and the balancing of different delay paths through the circuit is becoming a more important practice. There are several types of hazards [2] that can be classified as static and dynamic, or function and logic hazards. Function hazards are inherent to the function being implemented and occur in any implementation, logic hazards are specific to a particular implementation of a function and could still occur if function hazards are avoided.

Function hazards are avoided by restricting the inputs transitions to single variable changes, which is the fundamental mode of operation. Logic hazards are mainly avoided by choosing a different implementation or adding redundancy to the network used. Logic hazards are apparent in circuits with reconvergent signals. Some hazards could be removed by equalizing delay paths in the circuit at hand.

Hazards and glitches could be catastrophic to an asynchronous circuit designer, since they would cause misfiring of different events and cause system failure. Fortunately the synchronous design paradigm alleviates those issues by giving enough settling time for all intermediate transient values called the setup time of a flip-flop or latch before the clock event, which brings that stable value to the outside world as output of a flip-flop or latch with no glitches. The unfortunate part of that is that all these glitches and hazards still cause unnecessary power consumption.

3 CIRCUITS STUDIED AND THE NEW FLIP-FLOP

The following flip-flop circuits are from [1][3]. They were built using the Cadence schematic capture Virtuoso tool and sized for minimum size to function correctly. From [1], we concluded that the worst case power consumption is not dependent on clock frequency or D-to-Q delay unless the setup condition is violated, i.e. the flip-flop changes the region of operation. Another conclusion from [1] is that the least power consuming flip-flops are the ones that really deserve to be compared to any new flip-flop, therefore this paper focuses on the least power consumption flip-flops from [1][3] and compares them to the new flip-flop that we have developed. We later perform more detailed simulations so that we can compare the different aspects of the flip-flop designs.

We next present each of the flip-flop circuits considered in this paper accompanied by a short description of each circuit. Figure 2, shows the Power PC master-slave latch that is one of the fastest classical structures. Its main advantage is the short direct path and the low power feedback. The large load on the clock will greatly affect the total power consumption of the flip-flop. This flip-flop is the transmission gate flip-flop, it has a fully static master–slave structure, which is constructed by cascading two identical pass gate latches and provides a short clock to output latency. It does have a poor data to output latency because of the positive setup time. Moreover, it is sensitive to clock signal slopes and data feed through. This adds another concern when using it.

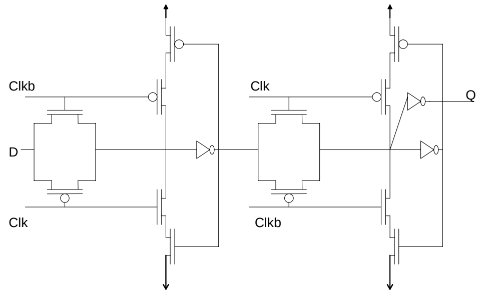

Figure 2 Power PC 603 MS latch.

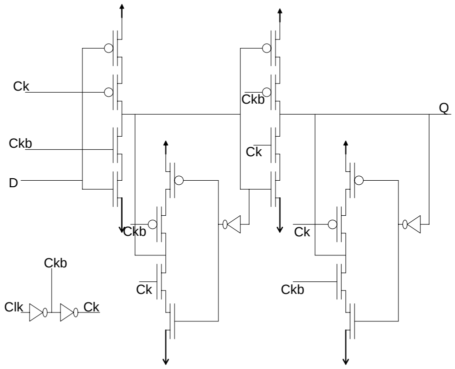

Figure 3 Modified C^{2}MOS latch.

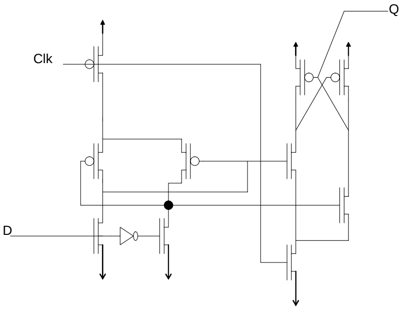

Figure 4 DSTC flip-flop.

Figure 3, shows the modified standard dynamic C^{2}MOS master-slave latch that has shown good low power features, like small clock load and low power feedback. The modified C^{2}MOS is also robust to clock signal slopes.

Figure 4 shows the dynamic single-transistor-clocked (DSTC) flip-flop that suffers from substantial voltage drop at the output due to the capacitive coupling effect between the common node of the slave latch and the floating output driving node of the master latch. This effect takes place at the rising edge of the clock and causes an increase in delay and short circuit power consumption in the slave latch which could dominate the dynamic power consumption. The capacitive coupling, floating node and data input signal glitches result in these flip-flop having lower driving capabilities than the rest of the flip-flop circuits. This should be taken into account by adding the power consumption of the dummy loads into the power

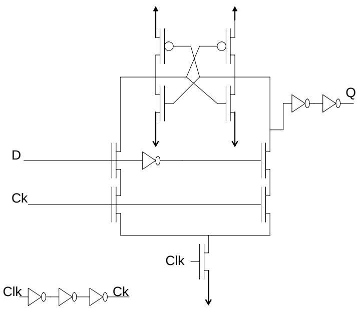

Figure 5 New ETL flip-flop.

measurements and calculations.

The new edge triggered latch proposed in Figure 5 is a modification of the K6 ETL [3] by replacing the jam-latches and adding the pull down transistors to create cross coupled inverters. Without the pull down transistors (of the back to back inverters) the flip-flop is still functional but the internal zero nodes suffer from cross coupling with the clock signal which causes an increase in the dynamic power consumption and reduction in the noise margins. The output inverters are not needed for correct circuit operation but are placed for general loading situations and to guarantee the internal storage node is not exposed to the output load directly which is a recommended practice for flip-flops and latches.

4 Simulation: Model, Method, and Results

4.1 Simulation Model

All flip-flop circuits were sized for minimum size transistors of a 90nm technology initially and sized up iteratively for correct functionality. Performance was not a sizing criterion and the idea behind this is that our goal is lowest power possible, which implies reduction in loading effects. We did see failures at certain clock frequencies and that is the only performance sizing effort that was done, bettering of performance was not one of our goals in this paper. For a general design situation, the inputs were driven with minimum size buffers and the outputs were captured after a minimum size buffer stage as well.

All the circuit power consumption was included in the measurement of maximum power, due to the fact that this is the real maximum power that will be consumed if the circuit is used as part of a system. This is to account for the effects of the inputs, the driving capabilities and glitches, if any, on the flip-flop outputs.

Figure 6 shows the basic model used for all the simulations done on the circuits presented in this paper.

All the numbers and results presented here on are from these simulations under the following conditions. The simulations were done at 25 degrees Celsius, with a 1.2 volts V_{dd} power supply and at the target process corner. We simulated all circuits at 50MHz. With relative schmooing of the data input relative to the clock with specific increments of setup time (0, 0.25, 0.5, 0.75, 1, 1.5, 2, 2.5, 3, 3.5, 4, 8, 12, 16, 20ns) which give more granularity of simulation points at the region of operation change. This enabled the measurement of the worst Clock-to-Q and Data-to-Q delays and power. In total there were $4 \times 15 = 60$ simulations to get the results and many more for design, debug purposes, and sizing iterations.

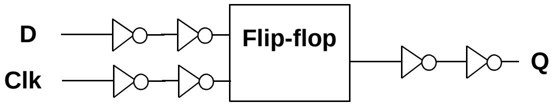

Figure 6 Simulation setup for flip-flops.

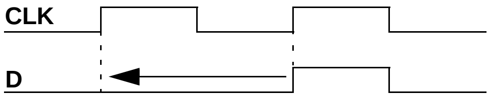

Figure 7 Simulation method.

4.2 Simulation Method

Ideally, for any flip-flop, a designer would like to sweep the clock and data inputs relative to each other through the whole range, which in this case would be a whole clock cycle. Since most of our models and simulators are sample based, which implies a discrete instant of time, the sweeping will have to be at discrete times. This leads to lower accuracy, but again the smaller the sweep increments the higher the accuracy. This point will be illustrated in the simulation results later.

To Simulate each and every flip-flop, we swept the data input edge relative to the latching edge for the edge triggered flip-flop circuits as shown in Figure 7. We did this on multiple iterations to identify which windows are the windows where the flip-flop changes the region of operation. Then we used smaller increments in the windows which need more investigation. As mentioned above, the sweeping for 50MHz was done for a rising data input edge and a falling one as well and choosing the worst values.

4.3 Simulation Results

In this sub-section we present the delay and power simulation results for the presented flip-flop circuits. As mentioned above we swept the data input relative to the clock and the data to output (D2Q) delay behavior of the flip-flops are shown in Figure 8. The figure shows how the flip-flops (each flip-flop number matches the corresponding figure in Section 3) follow the curve shown in Figure 1. It is worth noting that in the failure region the output of a typical flip-flop does not follow the input. The reason for the data points given there is the way we trigger the capture of the delays in HSPICE. The delay @20ns is identical to the one @0ns because the event of capturing the delay happens one clock cycle later. The optimal setup time for the new flip-flop (flip-flop #6) would be 1ns, where the D2Q is minimal. All other flip-flops exhibit the same behavior with different corner delays as shown in [1].

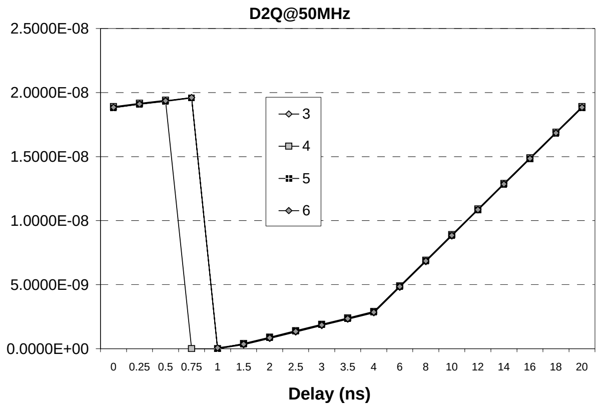

Figure 8 D2Q for flip-flops compared.

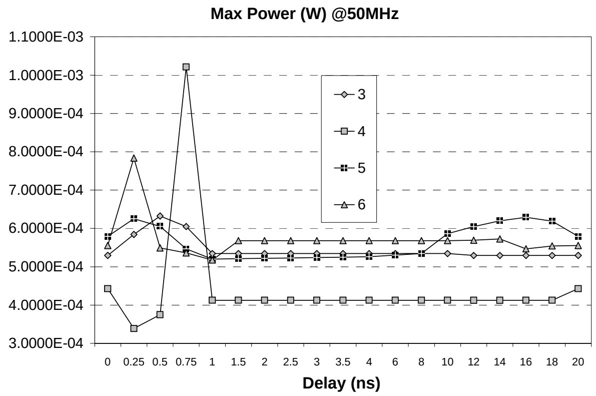

Figure 9 Maximum power for the considered flip-flops.

The new flip-flop exhibits the typical behavior of flip-flops used for low power applications. By comparing the new flip-flop to the other flip-flops, we can observe some important points. Flip-flop #4 (Figure 3) has a better setup time (0.75ns) than the others which are identical (1ns). Again the data points in the failure region are because the latching happened in a later clock cycle. All flip-flops have the same D2Q behavior and are comparable.

Figure 9 illustrates the max power (in Watts) consumed in the flip-flops for all the setup instants used for sampling. The figure shows that in the failure region the power might be higher than expected for any other operating delay point. By comparing the new flip-flop to the other flip-flops, we note that flip-flop #4 exhibits unexpected power consumption at its optimal setup delay point. The other flip-flops exhibit the same behavior but the new flip-flop does not. Flip-flop #5 exhibits another interesting unexpected property which is the increase of power consumption when the setup time is increased from 8ns to 10ns. These two observations are worthy of further investigation.

5 DISCUSSION AND CONCLUSIONS

If we would consider the number of transistors as a rough metric of area, given that minimizing the size of transistors was one of the main goals, then we can compare the area of the flip-flop designs. The best area is flip-flop #5 (12 transistors) and the worst is flip-flop #4 (24 transistors). Flip-flop #3 uses 16 transistors and the new flip-flop uses 21 transistors. Since the transistors are quite small in area, the effect of the difference in the number of transistors is diminished in larger designs where the flip-flops and latches are a lower percentage of the gate count due to the large combinational logic blocks used to perform the main function needed.

From the above observations and discussions we conclude that it is very important to increase the number of samples where the flip-flops are being simulated to get better accuracy. Another conclusion is that flip-flop #4 has maximum area, while best in power overall, still is not best at its optimal setup time. The introduced flip-flop though not the best overall power is the best power consumption at its optimal setup delay with moderate area.

We conclude this paper by bringing up an important set of guidelines which are the corner-stone for a low power flip-flop design methodology and low power flip-flop simulation in general. These are obtained from the lessons learned from all the experiments conducted in this paper.

***Method of design*:**

a. Minimize number of transistors.
b. Minimize load on clock.
c. Make internal nodes at full swing & not float at any time.
d. Minimize switching including glitching.
e. Remove redundancy except if used to remove glitching or reduce leakage.
f. Minimize size of transistors.
g. Go for functionality as priority while iterating for design and sizing above.

It is worthy to note that most of the above items are quite complex to accomplish and need a lot of insight and trial and error iterations to be able to reach these goals.

***Method for simulation*:**

a. Use a realistic model i.e. proper loading on outputs and driving sources on inputs.
b. Use realistic inputs' stimulus to capture the metrics you need to measure.
c. Simulate with coarse granularity to get the best functionality with minimal number of transistors and sizes.
d. Use a small step size in your HSPICE simulation. This helps in getting better accuracy.
e. Go back, analyze and redesign any irregularities in the trends of flip-flop behaviors.
f. Simulate for finer granularity at the corner delay values to gain more insight. This would increase the accuracy dramatically.

The above mentioned guidelines are more like an art than a methodology, an experienced design engineer would identify with the mentioned rules and would be able to direct his or her design to converge to the design goals (performance, power consumption, or area).

In summary, low power design for combinational and sequential circuits is an important field and gaining more importance as time goes by and will stay an important area of research for a long time. We have presented a new flip-flop design and compared it to competing low-power high performance flip-flop designs. Our experimental results enabled us to identify the trade-offs of existing flip-flop designs and helped us establish a set of guidelines for the design of low power and high performance flip-flop circuits.

REFERENCES

[1] A. Sayed and H. Al-Asaad, "Survey of low power flip-flops", to appear in *Proc. International Conference on Computer Design* (*CDES*), 2006.

[2] E. J. McCluskey, *Logic Design Principles*, Prentice Hall, 1986.

[3] V. Stojanovic and V. G. Oklabdzija, "Comparative analysis of master-slave latches and flip-flops for high-performance and low-power systems," *IEEE Journal of Solid State Circuits*, Vol. 34, pp. 536-548, April 1999.

[4] A. P. Chandrakasan, S. Sheng, and R. W. Brodersen, "Low-power CMOS digital design", *IEEE Journal of Solid-State Circuits*, Vol. 27, pp. 473 - 484, April 1992.

[5] P. Zhao, T. Darwish, and M. Bayoumi, "Low power and high speed explicit-pulsed flip-flops", *Proc. Midwest Symposium on Circuits and Systems*, 2002, pp. 477-480.

[6] R. H. Katz, *Contemporary Logic Design*, Benjamin/Cummings Publishing Company, Inc. 1994.