Circuits for Measurement of Flip-Flop Performance Variability

By Kenneth Duong

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in partial satisfaction of the requirements for the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee:

Professor Borivoje Nikolić
Research Advisor

(Date)

* * * * * * *

Professor Elad Alon
Second Reader

(Date)

Contents

1 Introduction
- 1.1 Increased Impact of Variability on Integrated Circuits
- 1.2 Recent Efforts to Study Variability
- 1.3 Impact of Variability on Clocked Storage Elements
- 1.4 Circuit Design for Variability

2 Flip-flop Timing Parameters
- 2.1 Definitions
- 2.2 Test Structure Resolution Requirements
- 2.3 Existing Bodies of Work in Delay Measurement

3 Variability of Flip-Flop Building Blocks
- 3.1 Purpose
- 3.2 Implementation Details
- 3.3 Expected Results
- 3.4 Choice of Flip-Flop Building Blocks
  - 3.4.1 Isolated Gate Structure Effect on Timing Parameters
  - 3.4.2 Composite Gate Structure Effect on Timing Parameters

4 Flip-Flop Delay Measurement Using Ring-Oscillators
- 4.1 Setup Time Ring-Oscillator
  - 4.1.1 Concept
  - 4.1.2 Implementation Details
- 4.2 Clock-to-Output Delay Ring-Oscillator
  - 4.1.1 Concept
  - 4.1.2 Implementation Details

5 Absolute Delay Measurement Circuit
- 5.1 Concept
- 5.2 Implementation Details
  - 5.2.1 Switch Mismatch
  - 5.2.2 Delay Measurement Using Edge Detector
  - 5.2.3 On-chip Variable Delay Element
- 5.3 Timing Parameter Measurement
  - 5.3.1 Setup Time Measurement
  - 5.3.2 Clock-to-Output Delay Measurement

6 Conclusion
- 6.1 Summary of Expected Results
- 6.2 Future Work

1. Introduction

1.1 Increased Impact of Variability on Integrated Circuits

The continuous scaling of integrated circuits over the past 30 years has presented circuit designers with a number of technological barriers to overcome. Resolution enhancement techniques such as optimal proximity correction and immersion lithography were developed in order to scale to deep-submicron devices with minimum feature sizes near or less than the wavelength of light used in lithography. While these and other advancements in process technologies have enabled continuous shrinking of critical dimensions, the variations in these process steps have not scaled accordingly. In addition to the growing sources of process variation, the larger number of transistors per die further increases the likelihood of more extreme cases occurring. As a result, process variability may now be the limiting technological barrier to further scaling.

1.2 Recent Efforts to Study Variability

Because of the increasing impact of process variability on devices, circuit designers face the challenges of identifying the sources of systematic and random variations and finding circuit techniques to mitigate them. For example, a study of a 90nm process showed that varying the proximity of the nearest poly-silicon line to a transistor gate caused a 10% systematic layout-dependent gate-length variation [1]. After this source of systematic variation was identified, restricted design rules on poly-silicon proximity were enforced and process and optimal proximity correction (OPC) adjustments were made.

As a result, a similar study of a 45nm process showed only a 2% systematic gate-length variation due to density [2].

The impact of variability on SRAM has also been explored extensively. SRAM is of particular interest because of the need for high yield and the large percentage of die area dedicated for SRAM-based cache memory. Because of the large array sizes, failures are statistically more likely to occur - the worst-case read and write margins decrease with increasing array size since an extreme in process variation is more likely to exist in a larger sample. Furthermore, SRAM cells are typically designed with minimum-sized transistors, which are affected the most by process variability. As a result, there have been a number of recent studies that either characterize or propose ways to compensate for the effect of process variability on SRAM functionality [3-5].

1.3 Impact of Variability on Clocked Storage Elements

Flip-flops and latches are another group of functional blocks whose performance could be greatly affected by the increase in variability of modern processes. Like SRAM arrays, flip-flops also represent a large percentage of chip area. Even conservatively estimating that flip-flops take up approximately 10% chip area, the increased transistor density at modern technology nodes means that large chips such as microprocessors or GPUs will hold at least a million flip-flops. As designing for the variability of SRAM noise margins is important for chip yield, variability in the timing overhead of flip-flops directly impacts the fastest achievable clocking rate in a pipeline, since uncertainties in flip-flop timing parameters require the assignment of extra safety margins. Thus, finding

ways to limit the variability of flip-flop timing parameters is beneficial in terms of both performance and reliability.

1.4 Circuit Design for Variability

One solution for controlling process variability in digital circuits, shown in Figure 1, is to use regularity in the layout of standard cells to greatly reduce systematic layout-dependent variations [6]. The use of regularity is not a new approach as analog circuit designers have already explored similar techniques to control mismatch between transistors as well as other circuit elements [7]. However, because of the driving economic benefit of scaling digital circuits, the area and performance penalty associated with regularity cannot be as easily accepted. Thus, while compromising area in order to control variability either with restricted design rules or uniform building blocks, may be the long-term solution to the variability problem, there is still a need to find design techniques at the circuit level that can control variability without drifting from Moore's Law.

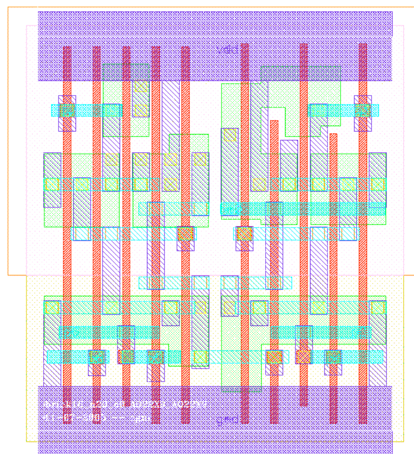


Figure 1. Example of regular geometry standard cell layout [8].

This project aims to measure and analyze the impact of flip-flop topology on variability. To keep the scope of this project manageable, we only consider various master-slave type flip-flops. However, the experimental structures developed here could easily be extended to other flip-flop topologies as well. Since master-slave flip-flops are complex structures consisting of many different transistor configurations, we expect that their timing parameters may be some composite function of these different blocks. Thus, to have a more complete understanding of how timing parameters are determined, it is necessary to study the individual building blocks first. With an understanding of the relative susceptibility of these blocks to variability, we can then study the composite effects of different blocks in master-slave flip-flops. The ultimate goal of this project is to propose general guidelines on what types of transistor structures are vulnerable to process variability, specifically for flip-flop design and more generally for digital circuit design in deep sub-micron technologies.

2. Flip-Flop Timing Parameters

2.1 Definitions

In order to get a better idea of the test structures that need to be designed, we must first define the timing parameters of interest. Setup time and clock-to-output delay are the most important quantities of interest since both of these timing parameters contribute directly to the overhead of a pipeline stage. Hold time constraints are only relevant in tight race conditions and can generally be margined to be insignificant. Thus, for the purposes of this study, hold time will be neglected.

There are several definitions for setup time and clock-to-output delay that should be considered in designing this study. One interesting view combines these two parameters into a single metric [1]. As shown in figure 2, setup time and clock-to-output delay can be defined as the values of $t_{D\text{-}CLK}$ and $t_{CLK\text{-}Q}$ for which the overall overhead $t_{D\text{-}Q}$ is minimized.

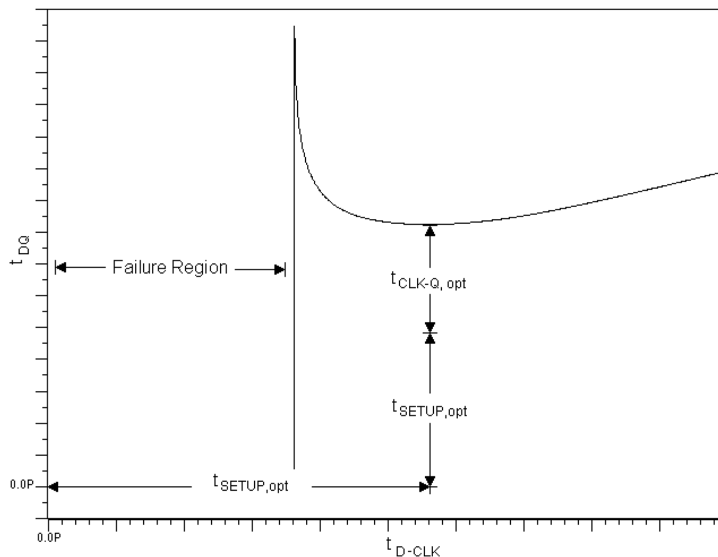


Figure 2. Optimal setup time and clock-to-output delay simulation.

While this definition is in theory the most relevant to the quantity of interest, the minimum overhead a flip-flop represents in a pipeline stage, it is difficult to measure accurately on-chip since there is no triggering event at the point of interest. Also, there is more to be gained by isolating the two timing parameters and studying them independently of each other.

The conventional definition of setup time is the latest allowable data arrival with respect to the relevant clock edge that still samples the correct data[1]. This definition allows for much more feasible test structures since failure to capture the correct data marks the setup time of interest. Clock-to-output delay can also be defined as the delay between the capturing clock edge and the transition to the correct output for a large setup time. Using these definitions, setup time and clock-to-output delay were simulated for a standard master-slave type flip-flop using a 45nm technology and are illustrated in Figure 3.

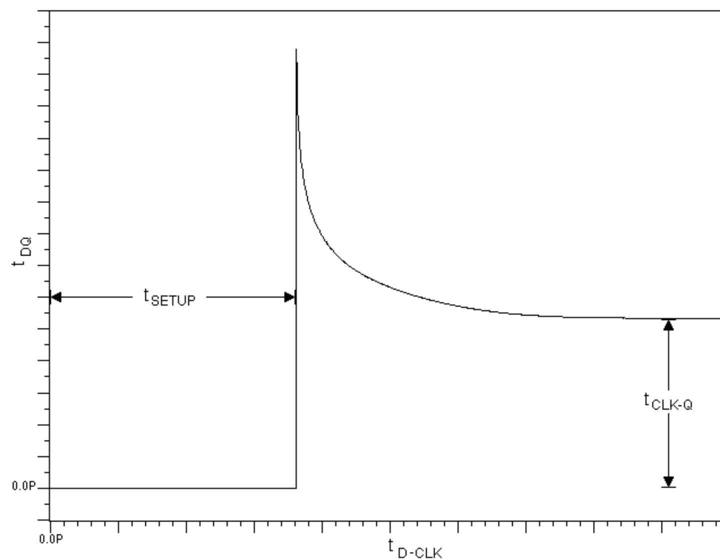
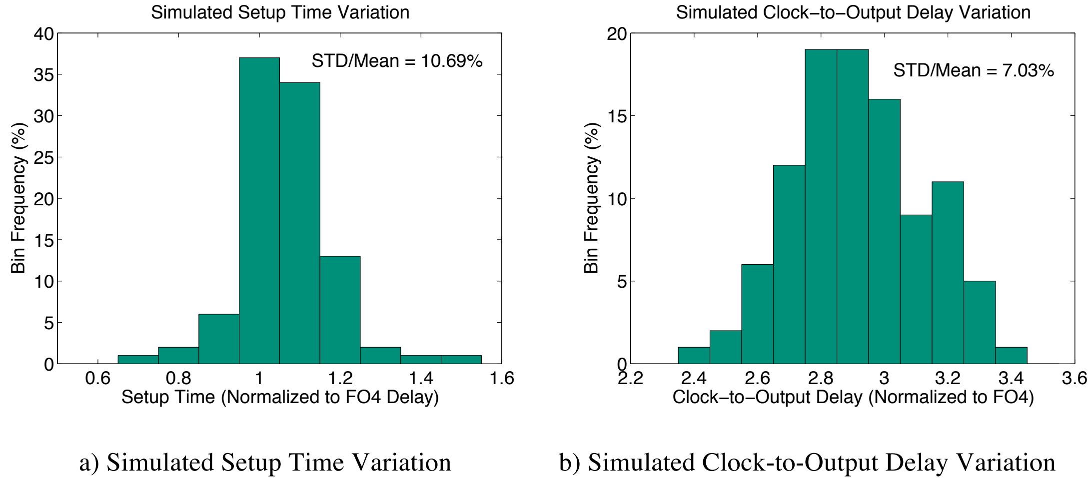


Figure 3. Conventional setup time and clock-to-output delay simulation.

2.2 Test Structure Resolution Requirements

Since we are trying to measure the variations of flip-flop timing parameters, the test structures we design must have high enough resolution in order for the collected data to be placed into a significant number of bins. To measure relative timing parameter variations, ring oscillators have been shown to resolve very small differences in delay [1]. By forming ring oscillator structures out of flip-flop setup times and clock-to-output delays, we can measure the relative variation of each of these parameters between different flip-flop topologies. While these oscillators provide an easy way to measure delay, they also average the random variation of several flip-flops in a local area and can prove difficult to form for the delays we are interested in, in particular, setup time. Thus, it may also desirable to have an absolute timing measurement circuit that can isolate the timing parameters of individual flip-flops.

Figures 4a and 4b show the results of Monte Carlo simulations of the setup times and clock-to-output delays of a standard master-slave flip-flop across a number of runs. To simulate setup time, the data-to-clock time is swept at half picosecond intervals until the flip-flop no longer captures the intended data. The last data-to-clock time that results in a correct output is the setup time. To simulate clock-to-output delay, the data is applied 200 picoseconds ahead of the clock signal. The clock-to-output delay is the time between the rising clock edge and the correct data output. Die-to-die (LOT) and within-die (DEV) variations were applied using a statistical device model with data fit to measured process parameter variations from foundry test-chips, and thus should give a reliable estimate of the actual process variability.



a) Simulated Setup Time Variation b) Simulated Clock-to-Output Delay Variation

Figure 4. Simulated timing parameter variation of a standard master-slave flip-flop.

2.3 Existing Bodies of Work in Delay Measurement

From these Monte Carlo simulations, we find that an absolute timing measurement structure would need to have a resolution of approximately 1/10 of a fanout-of-4 delay in order to resolve a standard deviation of the simulated variation of setup-time. The clock-to-output delay variation is significantly larger, and thus not limiting. While many researchers have published timing measurement circuits, none have had the level of resolution required for this measurement. The common limiting factor in resolution for these existing works in delay measurement is the variability in the test structures themselves. For example, the various conventional Vernier delay line implementations are limited at least by the variability in setup time of the flip-flops used to capture the propagated signals [10-11]. This suggests that in order to take reliable high-resolution measurements, some type of difference signal needs to be measured to calibrate out die-to-die variations in the measurement structure.

A timing measurement circuit that uses this strategy, in particular for flip-flop timing parameter measurement, is shown in Figure 5. The circuit achieves a resolution of approximately five picoseconds by measuring the delays in two different steps relative to

a common delay [12]. An on-chip reference generation circuit generates two reference signals that are skewed by steps of five picoseconds. One signal is passed through to a local skew generation circuit for generating different data-to-clk conditions while the other is used as a reference point for measuring each of the IO signals of the flip-flop under test. While this circuit is the highest resolution timing circuit that was found in the literature, it is still limited by the mismatch in delays through the multiplexors used to choose between the units under test, as well as by variability of the skew generation circuit. In section 5, a timing circuit is proposed that is able to circumvent or minimize most of these sources of error and achieve the required resolution.

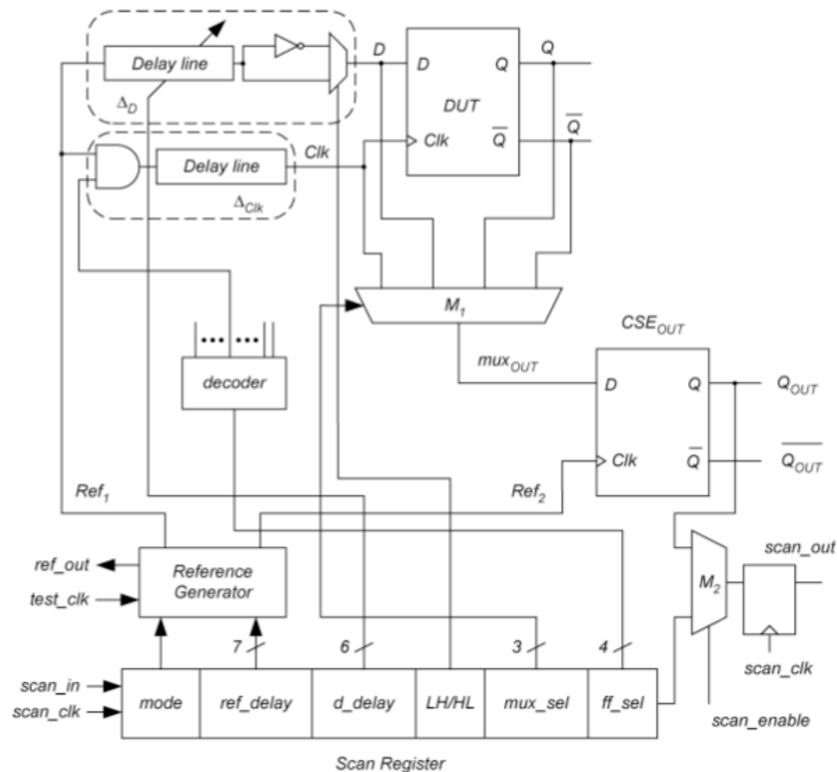


Figure 5. Difference-based flip-flop timing measurement circuit [12].

3. Variability of Flip-Flop Building Blocks

3.1 Purpose

In order to meaningfully construct an analysis of the effect of flip-flop topological choices on timing parameter variability, we need to first break down flip-flops into characterizeable units. As shown in Figure 6, a standard master-slave flip-flop is basically composed of a series of gated inverters.

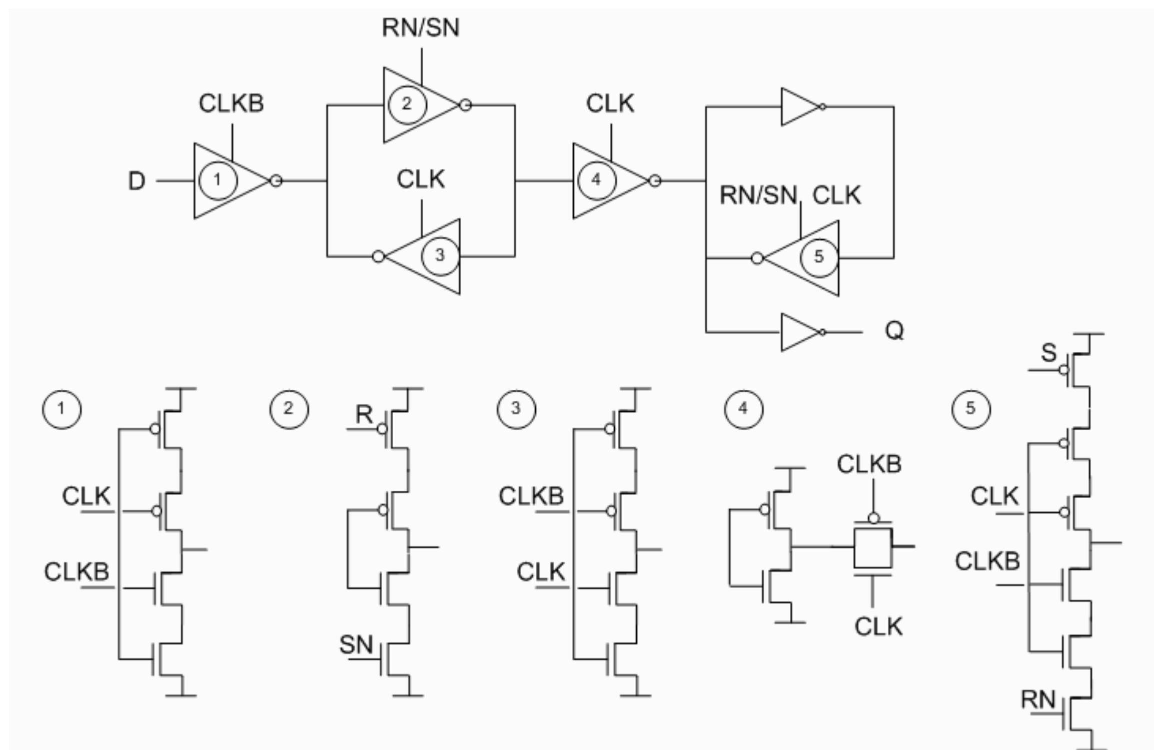


Figure 6. Standard master-slave flip flop.

These basic building blocks can be further dissected into individual transistor stacks with different switching behaviors. The underlying question this report is trying to answer is whether some of these structures, because of the modes of operation their transistors pass

through, are more susceptible to process variations than others. For example, if in a given structure, an NMOS transistor with a critical input spends significant time drawing current in a low V_{GS}, high V_{DS} mode - in which V_{TH} variations have a much stronger impact on the magnitude of the current – the pull-down delay of that structure will be much more variable. Rearranging the inputs into such a structure, or perhaps choosing a different, but functionally equivalent structure, could make the overall performance less variable while not sacrificing area or power. To illustrate this more clearly, we will analyze the operation of the pull-down stack of a NAND gate. The stack of two NMOS transistors has two pull-down delays of interest corresponding to two different input patterns:

(1) The top transistor is on and the bottom transistor switches on

(2) The bottom transistor is on and the top transistor switches on

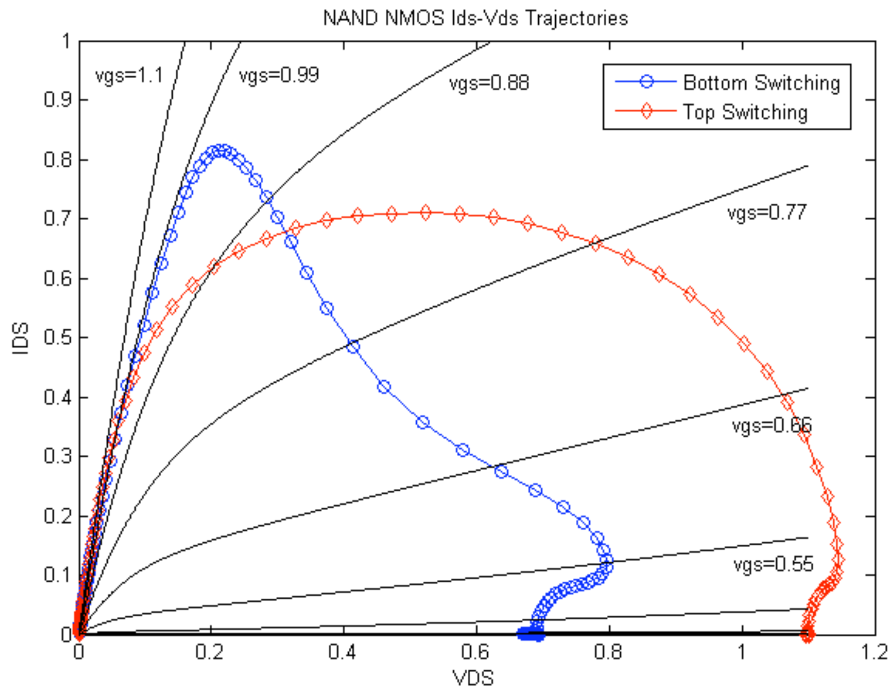


Figure 7. NMOS NAND I_D-V_{DS} trajectories.

The I_{DS}-V_{DS} trajectories for these cases are shown in Figure 7. The clearest difference between these trajectories is that the top-switching case passes through the more variable, velocity –saturated (low-V_{GS}, high-V_{DS}) region for a longer portion of its switching time (as shown by the larger number of points below V_{GS}=0.55V). As discussed above, this suggests that a NAND gate with a top-switching NMOS transistor will see higher relative variability in its pull-down delay.

3.2 Implementation Details

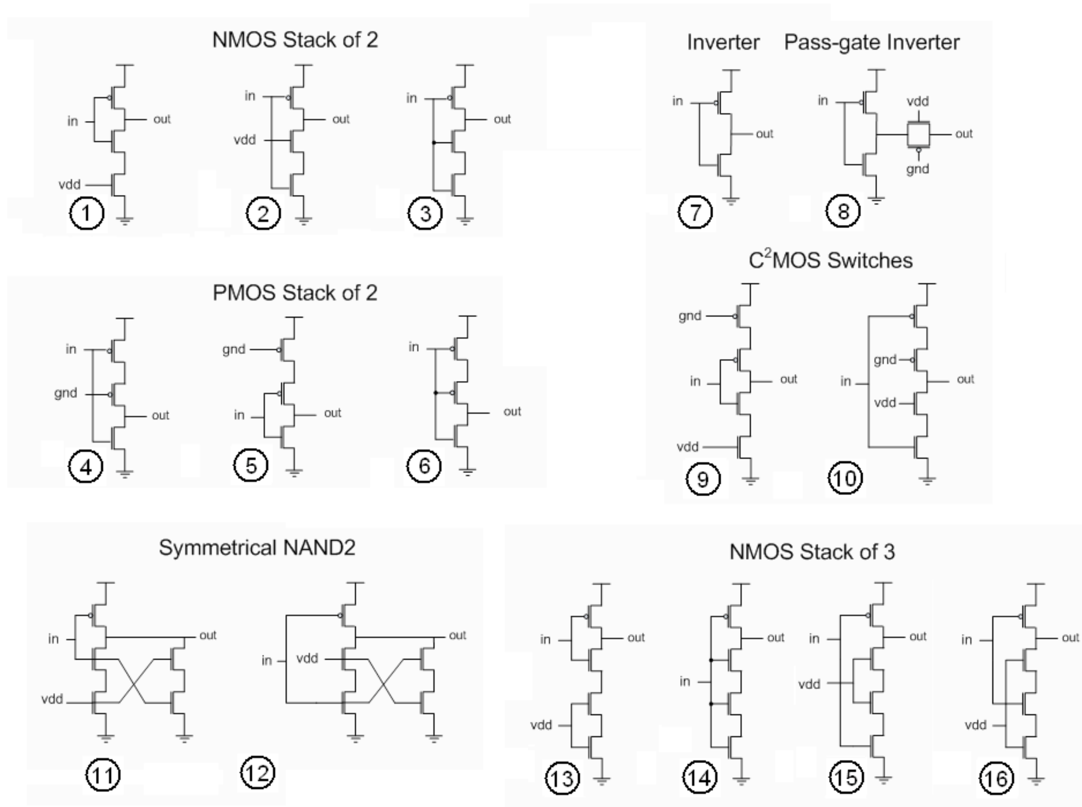


Figure 8. Inverting stack transistor structures.

Figure 8 shows the stack transistor structures that will be compared against each other in this study. To study the effect of switching position for NMOS transistors, three

inverting stages are created, each of which represents one of three input switching behaviors for an NMOS stack of 2. Each of these stages uses a single pull-up device so that all transistors can have identical layout (except for the metal-2 wires that actually connect the inputs in the appropriate switching configurations) to minimize systematic differences between them

Also, the layouts of the gates were kept very regular (all regular poly density, equal spacing between non-abutted gates) to avoid proximity effects and other avoidable systematic process variations. The different switching behaviors for PMOS stack of 2, NMOS stack of 3, and different gated inverters are also compared amongst each other. The regular inverter is used as a reference point for normalizing data, and the symmetrical NAND structures are used as a check for systematic variations that might occur because of layout asymmetries.

Since we only care about relative variation for these comparisons, ring-oscillator-based structures are used for the on-chip variability measurement. Random variations will be averaged out locally since several gate delays are added to produce a ring oscillator frequency, but variations between different locales will still be detectable. Also, the pull-up and pull-down delays of the different structures will be averaged, in some cases, undesirably. However, for those cases, the irrelevant network – the pull-up transistor in the NMOS stack of 2 comparisons, for example – is the same for all the configurations in a comparison group and thus is a common-mode error.

Each of the stack transistor structures is placed in tiles of the 13-stage ring-oscillator control structure shown in Figure 9. The number of stages must be large enough so that the delay (and thus contribution to variation) from the enabling NAND is

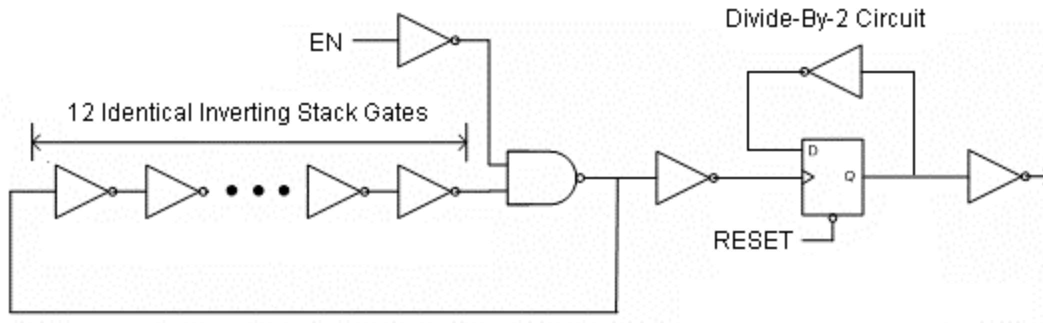


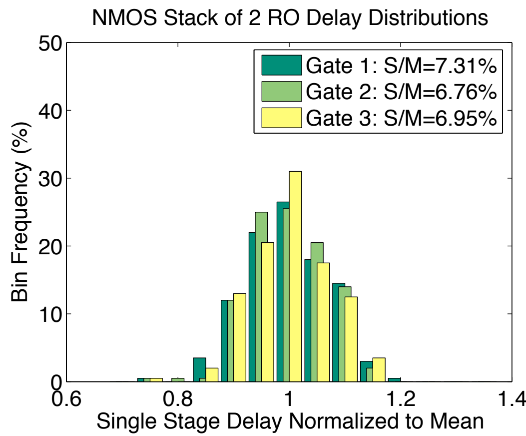
Figure 9. Ring oscillator control structure.

less than 10% of the overall ring oscillator period. In the ring oscillator array, row and column bits from a scan-chain choose a particular ring oscillator to multiplex out to a row frequency divider before being output to a pad. A local divide-by-2 circuit is located at each ring oscillator allows the use of a smaller number of stages by reducing the frequency of the signal being multiplexed out.

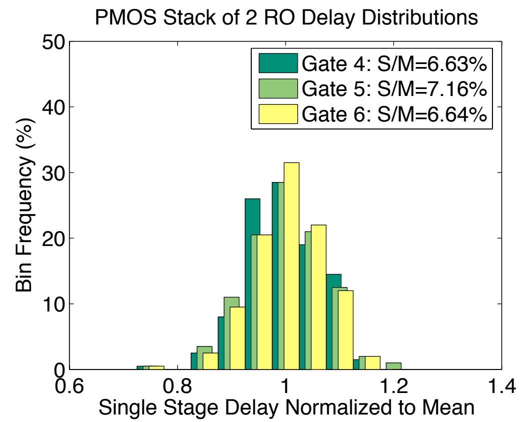
3.3 Expected Results - Monte-Carlo Simulations

From studying the stack of 2 NMOS I_{DS}-V_{DS} trajectories, we can expect that, in the other comparison groups, the transistor closest to the output node will trace through the low-V_{GS}, high-V_{DS} region longer (primarily due to a larger initial V_{DS}) and thus have a more variable delay. To verify this observation, each of the stack structures are placed in a ring oscillator and the oscillation period is measured across several Monte Carlo runs. Plots of the normalized variation (with respect to sigma/mean) of each of the comparison groups show us that the observation generally holds true (shown in Figure 10). The configurations in each relevant comparison group with a switching transistor closest to the output node (gates 1, 5, 9, and 13) all show higher relative variability than their counterparts (gates 2, 4, 10, and 15). In the cases for which all transistors in a stack are

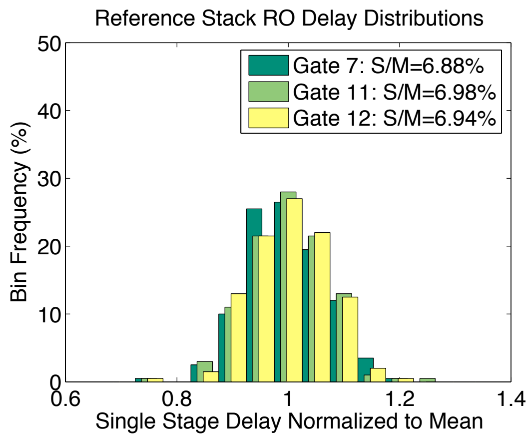
parts (gates 2, 4, 10, and 15). In the cases for which all transistors in a stack are switched on at once (gates 3, 6, and 16), the relative variation expectedly falls somewhere between the two previous cases. Also, as predicted, the reference symmetrical NANDs (gates 11 and 12) show the same relative variation. The simulation results from the various switches are the most relevant to the study of timing parameter variability and are discussed more comprehensively in the next section.



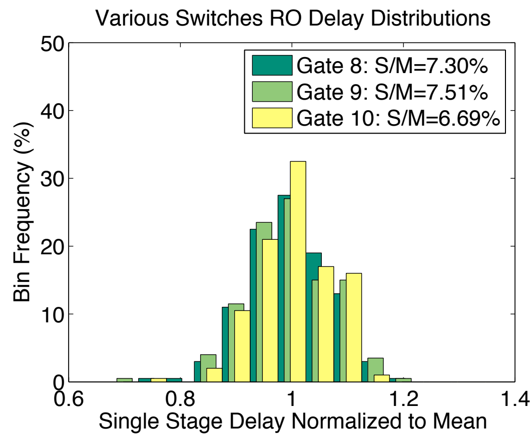
a) NMOS stack of 2 RO delay simulations.



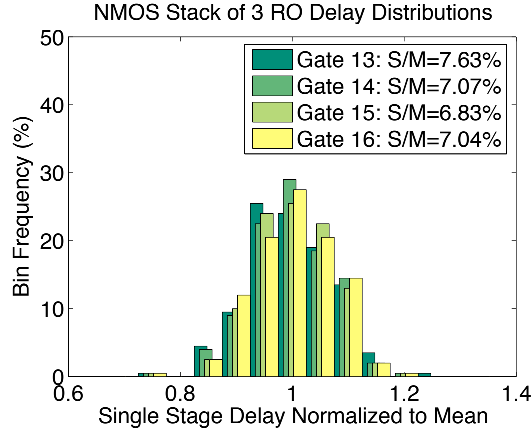
b) PMOS stack of 2 RO delay simulations.



c) Reference structure RO delay simulations.



d) Various switch RO delay simulations.



e) NMOS stack of 3 RO delay simulations.

Figure 10. Stack comparison group variation simulations.

3.4 Choice of Flip-Flop Building Blocks

With a better idea of which structures are less prone to variability, we can now dissect a standard master-slave flip-flop to study the effect of these structures on the variability of a flip-flop's timing parameters. Using the standard master-slave flip-flop of Figure 6 as the control, alternate flip-flops are generated by substituting functionally equivalent structures at various gate positions (shown in Table I). In order to control the number of permutations and maintain a fair comparison across flip-flops, guidelines were observed when making choices for which structures to interchange:

(1) hold time of the FF is negative

(2) only structures with first order effects on delays are of interest

(3) SET and RESET functions are implemented to maximize stacks

3.4.1 Isolated Gate Structure Effect on Timing Parameters

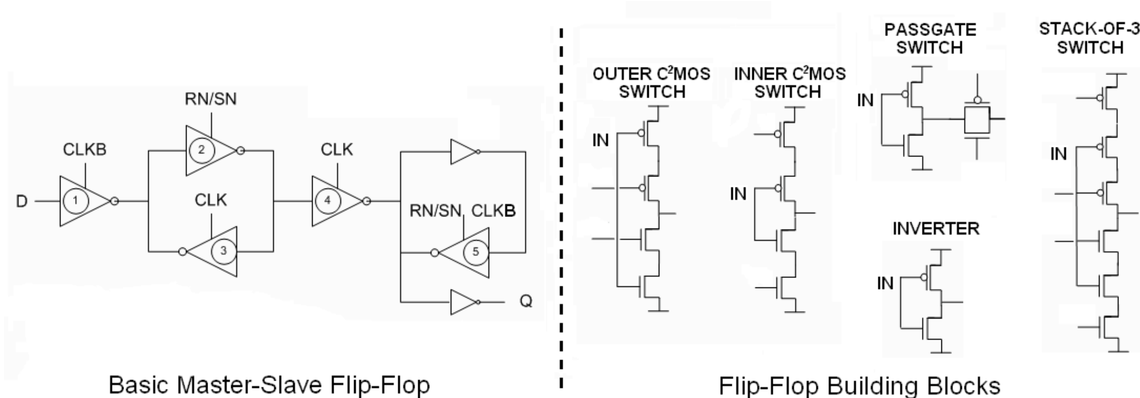


Figure 11. Breakdown of basic master-slave flip-flop and its interchangeable building blocks.

The first guideline constrains gate 1 shown in Figure 11 to be either the default control gate– a C^2MOS gate with the clock signal at the inner nodes of the stacks – or an inverter followed by a pass gate (flip-flop configurations 1 and 2, respectively - see Table I). This is because the arrival of the clock signal to gate 1 needs to cut off any coupling of the D input signal to the feedback structures of the latch. We also observe that setup time is only related to the behavior of the first latch since the minimum time for correct operation is set by how fast gate 1 can pass the input signal to gate 2 and close. Since gate 3 is not active until after the clock-edge arrives, its effect on setup time can be assumed to be a second order effect and thus negligible. Therefore, the three other types of stack structures, a regular inverter, an “inner C^2MOS switch,” and a “pass-gate switch,” are interchanged for the default gate 2 (flip-flop configurations 3, 4, and 5, respectively). Note that in order to use a regular inverter for configuration 4, gate 3 is forced to become a “stack-of-3 switch.” Also, as a result of the choices for gate 2, none of the configurations have an inner C^2MOS switch or a pass-gate switch at gate 3.

Similarly, for a non-critical setup time, clock-to-output delay is only related to the behavior of the second latch since the data feeding that latch is stable when the clock edge arrives. Again, the feedback inverters of the second latch have a secondary effect on clock-to-output delay since gate 5 is being disabled when the positive clock edge arrives. Thus, the default pass-gate switch at gate 4 can be replaced by an outer C^2MOS switch (flip-flop configuration 6) and gate 5 is not changed in any of the configurations.

TABLE I. FLIP-FLOP CONFIGURATION CHARACTERISTICS

	Gate 1	Gate 2	Gate 3	Gate 4	Setup (FO4)	CLK-to-Q (FO4)
1	OUT C2MOS	IN C2MOS	OUT C2MOS	PASS	1.08	2.14
2	**PASS**	IN C2MOS	OUT C2MOS	PASS	0.94	2.14
3	OUT C2MOS	**INVERTER**	**STACK3**	PASS	0.82	2.06
4	OUT C2MOS	**PASS**	OUT C2MOS	PASS	1.15	2.14
5	OUT C2MOS	**OUT C2MOS**	OUT C2MOS	PASS	1.24	2.14
6	OUT C2MOS	IN C2MOS	OUT C2MOS	**OUT C2MOS**	1.07	2.47
7	OUT C2MOS	**OUT C2MOS**	OUT C2MOS	**OUT C2MOS**	1.23	2.47
8	**PASS**	**INVERTER**	**STACK3**	PASS	0.76	2.06

3.4.2 Composite Gate Effect on Timing Parameter

The previous flip-flop configurations are used to study the effect of single gate deviations from the control configuration, and thus are measures of the isolated effect of a particular stack gate on a timing parameter. The composite effects of the variability of these blocks on the timing parameters also need to be tested. In order to do this, we must first define a metric that can fairly compare the variability of different flip-flops.

Recall that the main reason we want to decrease timing parameter variability is to minimize the overhead of the standard flip-flop. With this in mind, it does not make sense to switch to a topology with lower relative variability that is much slower than a flip-flop

with fast timing parameters and higher relative variability. Thus, the relative variation of a particular flip-flop's timing parameters to its mean is not really the primary quantity of interest. Rather, the metric for a flip-flop timing parameter should be some combination of the mean timing parameter plus its absolute variation. In order to ensure that 99.7% of the timing parameters are bounded, we propose an equation for the figure we try to minimize:

$$[1] \qquad T_{vtm} = T_{param,mean} + 3 * \sigma_{param}$$

As a result of this definition of the relevant metric to use for comparisons, the baseline for normalization should be area. In other words, it is irrelevant to resize the gates of the different switches such that they have the same delay if this equalized delay is significantly larger than the minimum delay achievable for one of the switch topologies. By keeping transistor size constant, we also control for differences in input load capacitance, as well as the approximate variation each transistor sees (from Pelgrom's model).

Using this variability tolerance metric to characterize each of the building blocks with transistor area kept constant, and assuming that the total timing parameter metric value will be the root sum square of the individual building block metrics, we can generate a flip-flop with predicted best-case timing parameters when considering variability tc

The plotted distributions of the results found in Section 3.3 for the FF building blocks are shown in Figure 12. From these results, the proposed metric suggests that, if possible, regular inverters should be used in the main delay paths that determine timing parameters, and functions that need to be implemented with stacks should be pushed into the feedback transistors. This makes the best-case flip-flop gates 2 and 3 an inverter and a stack-of-3, respectively. For gates 1 and 4, which are required to be either a pass-gate

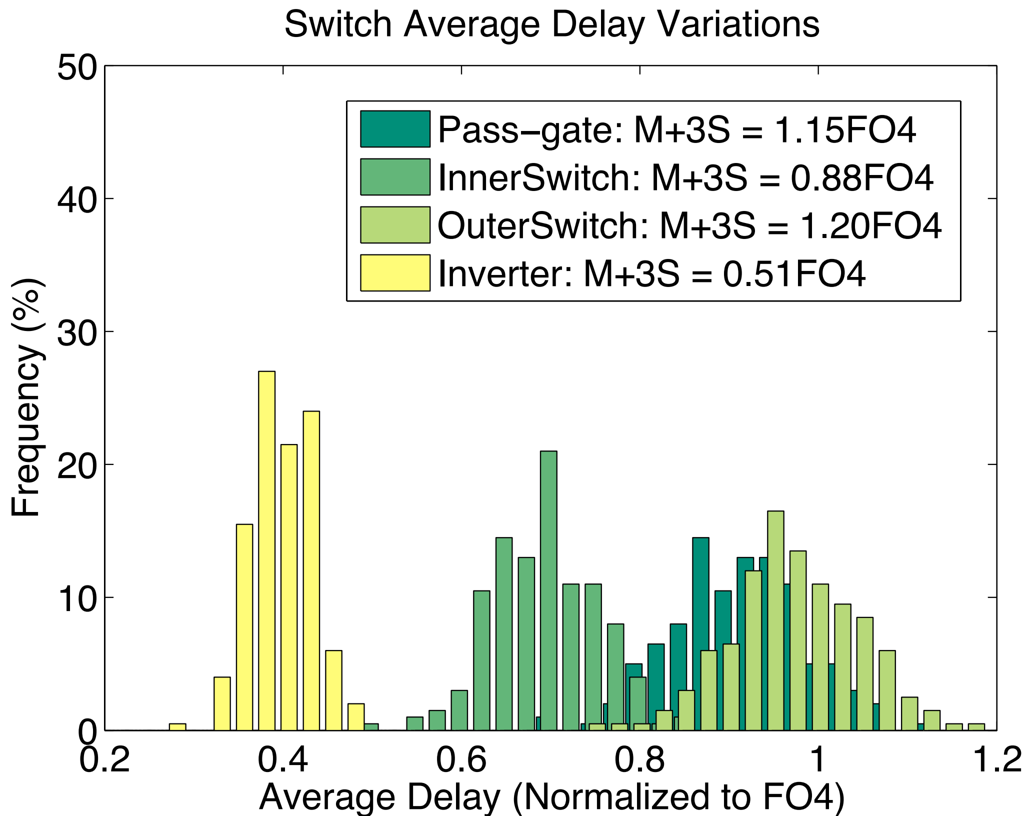


Figure 12. Comparison of various stacked switches using variability tolerance metric.

inverter or an outer C^2MOS switch by the guidelines, pass-gate inverters should be used in the best-case flip-flop since they have slightly better overall variability tolerance in simulation. This is likely due to the fact that, in comparison to the outer C^2MOS switch, the reduction of switching resistance (because of the parallel PMOS and NMOS combination of the pass-gate) outweighs the increase in internal node capacitance such that the mean delay is less while the absolute variation is about the same. Configuration 8 in Table 1 summarizes the blocks used for the predicted best-case flip-flop. Of all the building blocks used, we see that the outer C^2MOS switch has the best relative tolerance to variation (sigma/mean). Thus, we form configuration 7 out of just these building blocks as a

check to see if our line of reasoning holds true and this configuration has timing parameters with the best relative tolerance to variation.

Setup time and clock-to-output delay Monte Carlo simulations were run for each of the configurations of Table I. The proposed timing parameter metrics for each configuration are summarized in Table II, and generally agree with our predictions.

TABLE II. SETUP TIME AND CLOCK-TO-OUTPUT DELAY MONTE-CARLO SIMULATION (NORMALIZED TO FO4)

Description		Setup Time			Clock-to-Output Delay		
		Mean	σ	Mean+3σ	Mean	σ	Mean+3σ
1	Control	1.08	0.108	**1.40**	2.17	0.173	**2.69**
2	Gate 1 – Pass-Gate	0.94	0.086	**1.20**	2.18	0.164	**2.67**
3	Gate 2 – Inverter	0.77	0.087	**1.03**	2.11	0.164	**2.60**
4	Gate 2 – Pass-Gate	1.12	0.121	**1.48**	2.17	0.161	**2.65**
5	Gate 2 – Outer C²MOS	1.23	0.131	**1.63**	2.17	0.166	**2.67**
6	Gate 4 – Outer C²MOS	1.07	0.115	**1.41**	2.52	0.197	**3.11**
7	All Outer C²MOS	1.22	0.121	**1.58**	2.52	0.201	**3.12**
8	Best V.T. Metric	0.71	0.071	**0.92**	2.11	0.163	**2.60**

As expected, the flip-flops with replacement structures only in the master latch and thus aimed for studying setup time variation (configurations 2, 3, 4, and 5) all showed similar clock-to-output delay characteristics but different setup time characteristics. When gate 1 is changed to a pass-gate inverter, which had a better variability tolerance than the default gate in simulation, the overall setup time variability tolerance metric drops by 14%. Similarly, as predicted by the individual gate variability tolerance metrics, replacing gate 2 with an inverter and pushing the SET/RESET logic into a stack of 3 at gate 3 resulted in a better setup time, while replacing gate 2 with a pass-gate and outer C^2MOS switch resulted in worse setup time.

With respect to clock-to-output delay, flip-flop configuration 6 and 7, which both use outer C^2MOS switches as the control gate of the slave latch, have the worst clock-to-output delay metric overall, but unexpectedly, also have worse relative variation compared to the control configuration. The reason for this is that the default pass-gate inverter performs better than expected; when used in a flip-flop as the first gate of the slave latch, the latest arriving signal occurs at the input to the pass-gate as opposed to the input of the inverter. Thus, the inverting pass-gate structure that was compared against the other two switches does not characterize the same switching behavior that occurs at gate 4. Finally, flip-flop configuration 8 exhibits the best setup time and clock-to-output delay metric values, as expected.

4. Flip-Flop Delay Measurement Using Ring-Oscillators

As a first pass at studying the combined effects of building block variability on a flip-flop's timing parameters on-chip, we explore ring-oscillator based structures that can characterize the average timing parameters of several flip-flops in a local area without having to measure an absolute delay. While this only gives us a relative measure of the variability, the measurement results from these structures can still help support the general trends we observe and confirm the results from an absolute timing measurement circuit.

4.1 Setup Time Ring Oscillator

4.1.1 Concept

Since the setup time of a flip-flop is necessarily based on a relationship between the data and clock edge arrivals, it is very difficult to create a ring oscillator structure around a flip-flop that does not add too much variation between test structures.

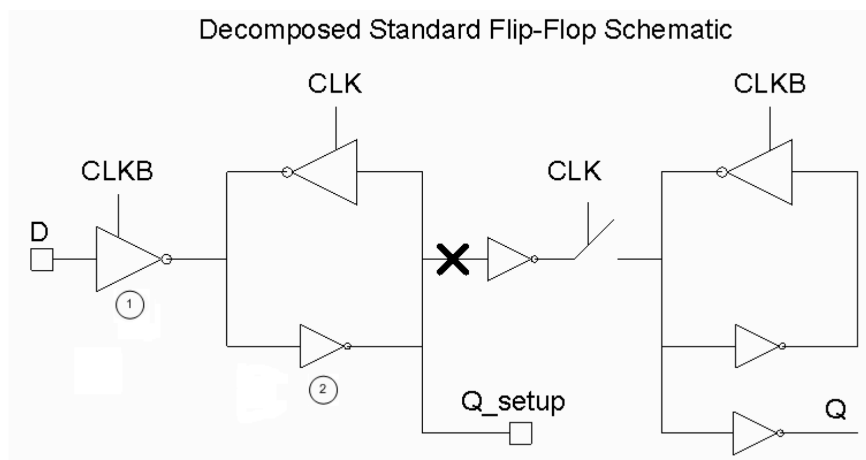


Figure 13. Flip-Flop schematic with proposed correlated setup-time delay.

The compromise is to create a ring oscillator that characterizes a delay through the flip-flop that is closely correlated to setup time. Failure occurs when a late change at the input, with respect to this clock edge, cannot propagate through gates 1 and 2 (shown in Figure 13) to be ready for latching at the clock edge. Thus, setup time of a flip-flop can be attributed to the delay through the master latch while the clock is low (prior to capture). To verify this assumption, Monte Carlo simulations are used to measure both the setup time and the delay through these two gates for the same control flip-flop in multiple runs. The results, shown in Figure 14, show that there is a good correlation between these two times.

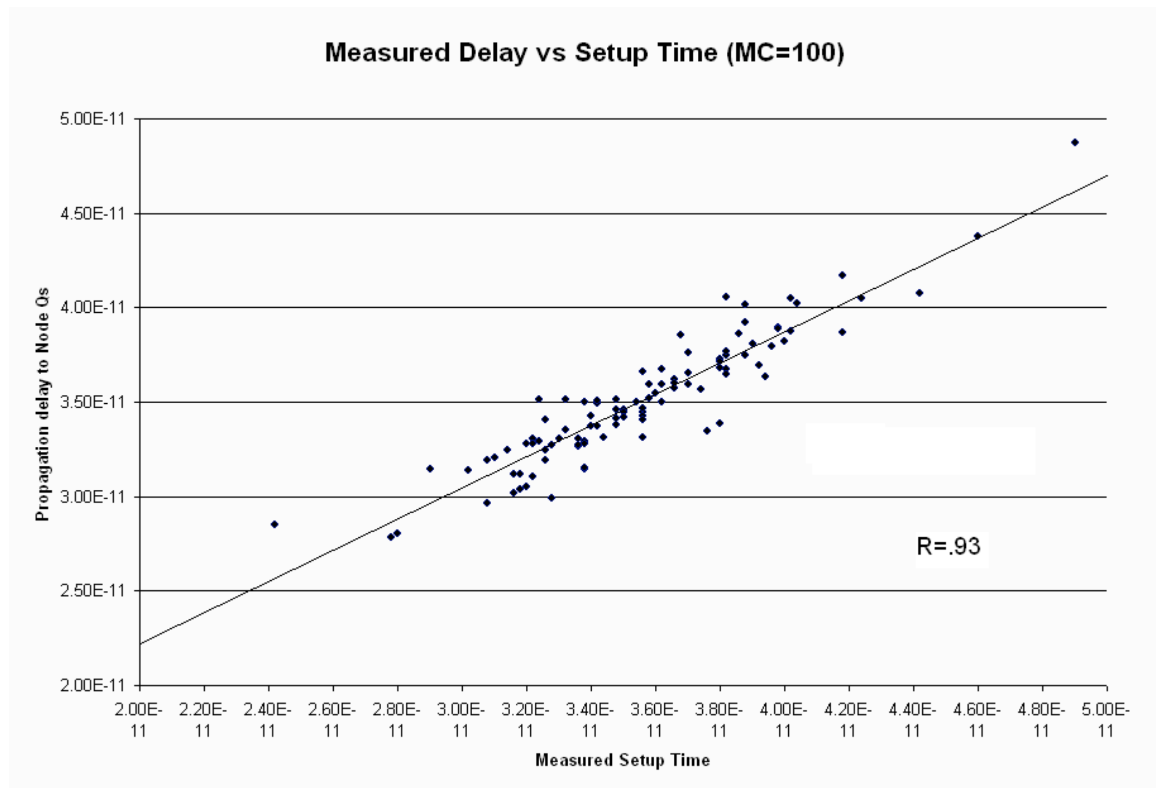


Figure 14. Correlation between simulated setup time and delay to node Q_S.

The correlation between the two simulated delays is good, but not excellent. This is because the propagation delay of gates 1 and 2, when the clock is artificially held low, is not affected the closing of gate 1 by the clock edge as it would be under actual setup time conditions. Also, if a flip-flop with its node Q_S accessible is used in a ring oscillator, the load that the second gate sees will not be the same load it sees in its natural environment. Still, since we are only interested in the relative variation of the different flip-flop configuration setup times, this experiment should give us a good idea of the composite effect of the building blocks on setup time.

4.1.2 Implementation Details

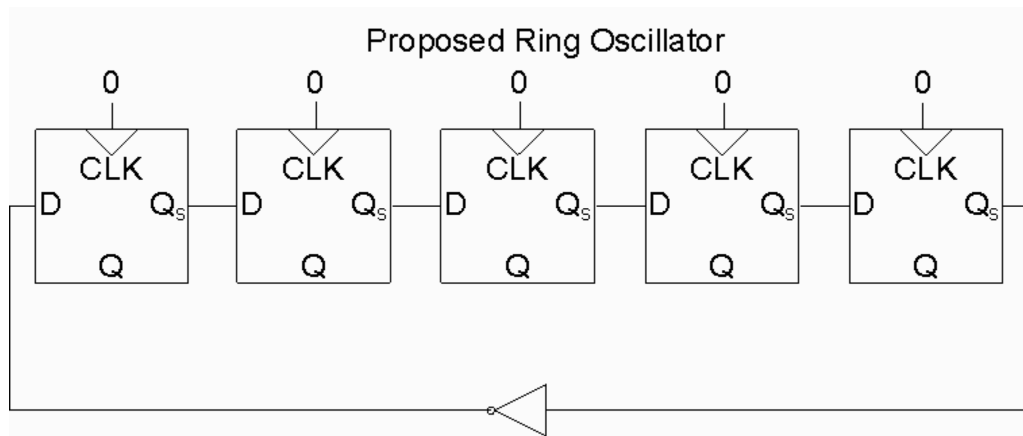


Figure 15. Schematic of setup time ring oscillator.

A schematic for the ring-oscillator built for the experiment is shown in Figure 15. A setup time ring oscillator is composed of five flip-flops of the same configuration and an inverter used to create an odd number of inverting stages (the derivative flip-flops are non-inverting). Each of the flip-flops has its clock input tied low to simulate the conditions similar to setup time failure. The minimum number of flip-flop stages is set by the

amount of variation to be tolerated by the inverter (chosen to be $< 10\%$) as well as by the maximum frequency that can be multiplexed through the test structure. The final array consists of 16 rows of tiles, with each tile containing a row for each of the eight flip-flop configurations described in section 3.4. The two outermost ring oscillators of each row and column should be ignored unless edge effects of the array are of interest.

4.2 Clock-to-Output Delay Ring Oscillator

4.2.1 Concept

Similar to the setup time ring oscillator, the goal of this structure is to approximate the average clock-to-output delay of a group of nearby flip-flops. While the setup time ring oscillator is formed out of flip-flops that must be slightly modified due to the conditions on setup time measurement, clock-to-output delay has less complicated measurement conditions. As a result, we are able to design a clock-to-output delay ring oscillator using unmodified flip-flops by taking advantage of the implementation of asynchronous set and reset (see Figure 16). The basic delay we are trying to measure is the delay between the rising edge of the clock and the rising or falling edge of the output for a given flip-flop with a long setup time. The cascade of falling edge output flip-flops would require an inverting gate since all of the flip-flops designed are positive-edge clocked.

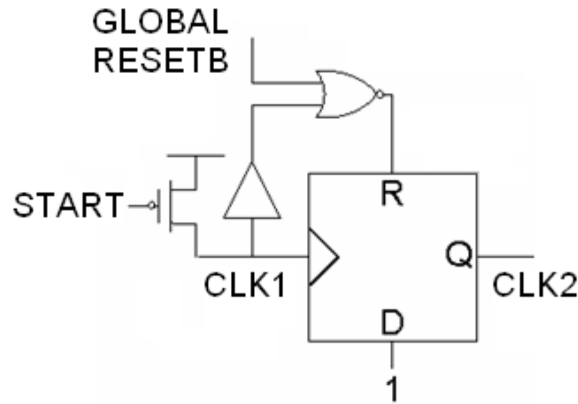


Figure 16. Single stage in proposed clock-to-output delay ring oscillator.

Thus, to keep this experiment simple, we only study clock-to-output delay for ‘1’-data. Holding the data input into a flip-flop at ‘1’, we assume that a positive clock edge can be generated into the flip-flop. After the positive clock-to-output delay of the flip-flop, its output will rise. The average of N clock-to-output delays can be created by cascading N stages of these flip-flops. However, in order to create a ring-oscillator with a characteristic frequency that can easily be measured, the output of the flip-flops must be reset at some point. The reset time after the rising clock edge of a given flip-flop is limited by the requirements on the next few flip-flops; the next flip-flop needs to have its clock signal high for long enough in order to generate the clock signal for the its successor. Thus, the reset time is required to be at least two clock-to-output delays after the clock arrival.

4.2.2 Implementation Details

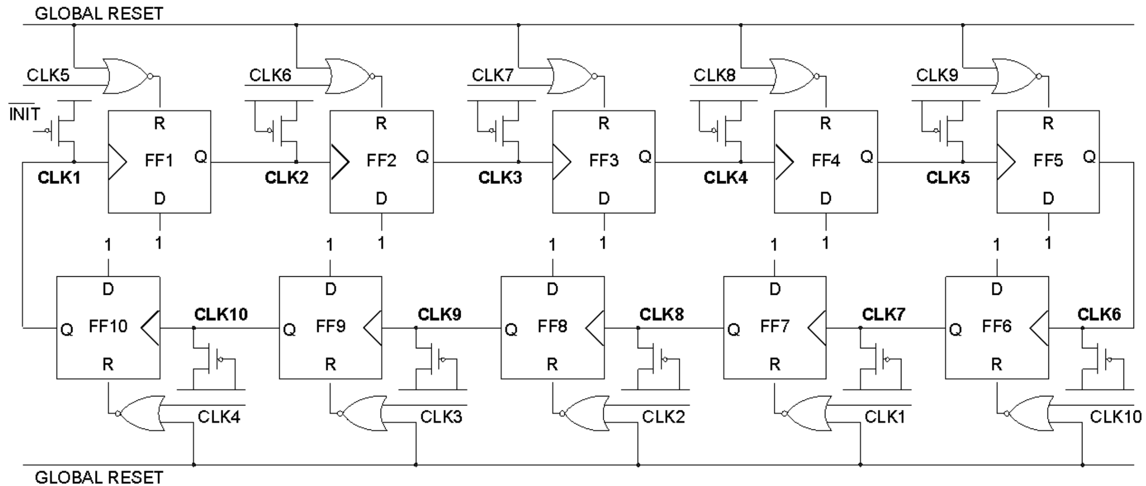


Figure 17. Clock-to-output delay ring oscillator schematic.

The final clock-to-output delay ring oscillator is shown in Figure 17. As a result of the requirements on the reset time, a relatively large number of stages is required; from simulation we found that ten stages guarantees successful oscillation across all corners. Instead of implementing buffers to generate the reset time after the clock edge, we simply use the clock signal of a flip-flop a few stages down the chain as the reset signal. This guarantees that a flip-flop output will not be reset until it has propagated far enough through the oscillator.

In order to start oscillation, an initial clock signal must be generated. A PMOS pull-up device is placed at each node to maintain symmetry, but only one of these pull-up devices is connected to a pulse generation circuit that generates the initial clock with a pulse of slightly less width than the duty cycle of a clock during regular oscillation. The test structure for outputting the ring oscillator frequency is the same as used for the stack structure ring oscillators.

Shown in Figure 18 is a functional simulation of the clock-to-output delay ring oscillator for the control flip-flop configuration. The START signal is generated by the pulse-generation circuit and starts the oscillation at the first flip-flop (FF1). CLK1 and CLK2 and are the input and output, respectively, of the first flip-flop.

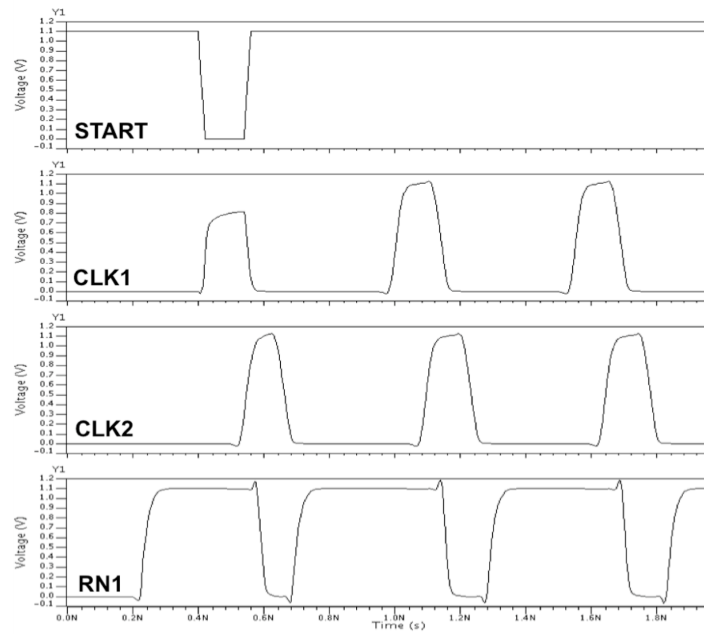


Figure 18. Functional simulation of clock-to-output delay ring oscillator.

5. Absolute Delay Measurement Circuit

5.1 Concept

The previous section outlined circuits that can give us measurements of average flip-flop timing parameters for nearby flip-flops. In order to compare variations die-to-die and wafer-to-wafer, however, it would be desirable to have an absolute delay measurement circuit. In section 3, we found that in order to obtain enough statistical data on timing parameters, the resolution of the circuit needs to be approximately 1/10 of a fanout-of-4 delay. While many techniques have been used in the past to perform time-to-digital conversion, the highest resolution circuits have used some sort of differential measurement in order to calibrate out variability in the measurement circuitry. Here, we propose a similar scheme that is able to limit errors in measurement down to the mismatch in delay between two nearby switches and thus meet the resolution requirements.

The concept of the proposed circuit is described in Figure 19. The basic premise is to create two delay paths, one path that traces a signal through the delay of interest, and another path that traces the same path, but with the delay of interest switched out. The underlying goal in building this circuit is to minimize the differences in the shared measurement circuitry as much as possible so that the difference in the two path delays is a good representation of the delay of interest.

5.2 Implementation Details

5.2.1 Switch Mismatch

In the circuit shown in Figure 19, the main source of mismatch between the two signal paths is in the switches used to choose between signal paths. In addition to using

layout techniques to minimize mismatch between the PMOS and NMOS transistors in the switches, we also keep the absolute delay through the switches as low as possible by driving the switches with large inverters. By minimizing the absolute delay through the

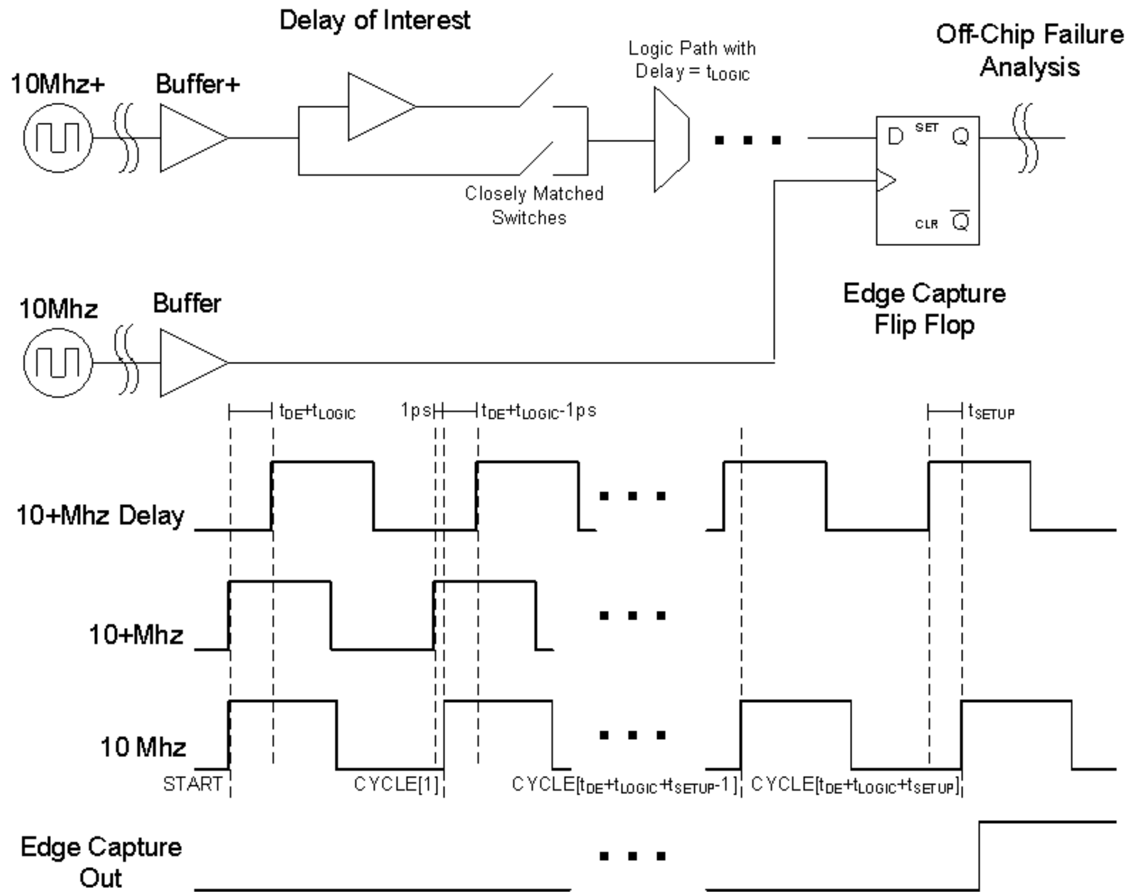


Figure 19. Illustration of concept for absolute delay measurement circuit.

switches, the variation of those delays is also kept small. For the final inverter and switch size used in the circuit, we simulated a mismatch of $3\sigma = 2.1$ps, which should give us enough resolution for the desired measurements.

5.2.2 Delay Measurement Using Edge Detector

Using a dual delay path scheme allows for less strict requirements on the actual delay measurement circuit since, outside of the immediate driver and switches near the delay element, the rest of the delay path is not restricted in sizing or length. The actual delay measurement scheme is also described in Figure 19. Two clock signals with a difference in period of less than a picosecond are generated off-chip with very little skew between them and are fed into two buffers. The fast clock (shown as the 10+MHz generated signal) is passed through the delay path into the data input of a edge detect flip-flop, while the slow clock (shown as the 10 MHz generated signal) directly drives the clock of the flip-flop. Because the initial clock edges start roughly at the same point in time, the fast clock, when propagated through the delay path, will not be able to meet the setup time of the edge capture flip-flop. The slow clock edges are counted until the fast clock is able to meet the setup time of the edge detect flip-flop and change its output. This clock edge count can then be multiplied by the difference in periods of the two reference clocks and approximately represents the delay through the signal path. Assuming that the initial clock edges line up, the delays that comprise this signal path are summarized here:

[2] $$N_{EDGES} \cdot (T_{SLOW} - T_{FAST}) = t_{SP1} = t_{BUFFER_DIFF} + t_{LOGIC1} + t_{DELAY} + t_{SW1} + t_{LOGIC2} + t_{SETUP}$$

where t_{BUFFER_DIFF} is the difference in buffer delays, t_{LOGIC1} is the logic delay before the actual delay of interest that includes control logic, t_{DELAY} is the delay of interest, t_{SW1} is the delay of the switch used to choose this particular signal path, t_{LOGIC2} is the logic delay of any succeeding control logic, and t_{SETUP} is the setup time of the edge capture flip-flop. This path delay is not very informative by itself, since none of the individual delays are

known. However, if a second switch is implemented such that the delay of interest can be bypassed, a second measurement can give us a way to calibrate out the irrelevant, unknown delays from the original measurement and isolate the delay of interest. Since we require a resolution on the order of one picosecond, care is taken to design the surrounding logic in such a way that rise and fall times of those signals after being driven through the switches are very similar to those generated by the delay element. Thus, assuming the rest of the signal path behaves exactly the same, the difference in the two measurements can be represented by:

[3] $$(N_{EDGES1} - N_{EDGES2}) \cdot (T_{SLOW} - T_{FAST}) = t_{DELAY} + t_{SW1} - t_{SW2}$$

Thus, assuming we can closely match the switches, the difference in the edge counts of the two measurements can give us a very accurate representation to the delay of interest.

Note an important assumption we made was that the two clock signals start at exactly the same time so that the two clock edge counts are truly representative of the total path delays. While clock frequencies can be guaranteed to be very accurate, delays of signal generators are less predictable. However, since the only real requirement is that the clock edge count for each measurement begins at the same time relative to some delay between the two input clocks (in the ideal case, we assumed this delay to be zero) a sync generator must be designed to have a reference point from which to measure. Another edge-detect flip-flop with its clock and data inputs directly driven by reference clocks will detect when the two clocks are within the flip-flop's setup time apart from each other.

5.2.3 On-chip Variable Delay Element

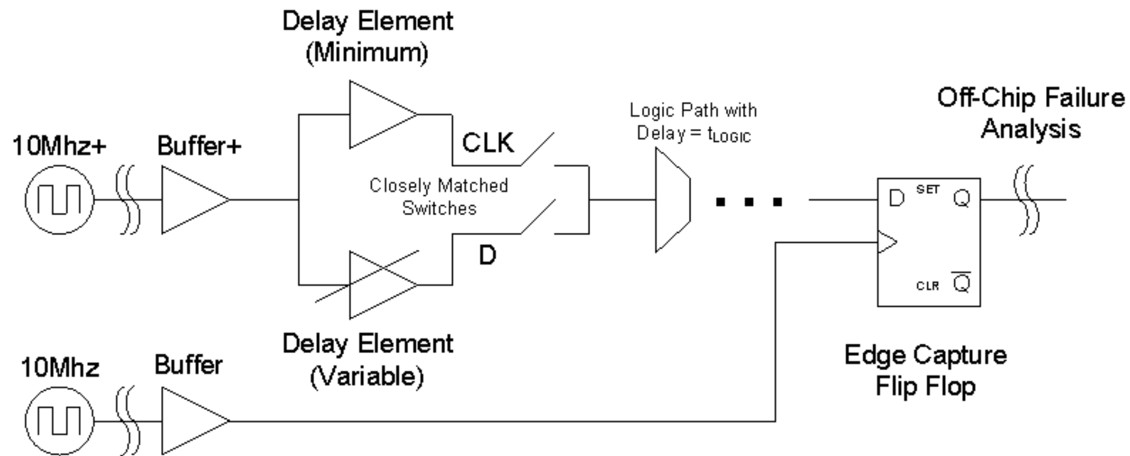


Figure 20. Dual path scheme with skew-generating variable delay elements

As long as measurements (clock edge counts) are taken with respect to the rising edge of this sync pulse, the actual start time of the clocks is not an issue.

Since we are interested in measuring flip-flop timing parameters, it is very necessary to be able to introduce fixed skews. For example, setup time needs to be measured by skewing the data and clock signals with respect to each other at fixed intervals. While there exist signal generators with high enough skew resolution for these measurements, we wanted to push as much of the measurement circuitry on-chip as possible. Thus, an on-chip variable delay element was designed to generate the data and clock signals for timing parameter measurement conditions. Because of these conditions, as well as resolution requirements, the delay element is required to generate anywhere from 10ps to 150ps

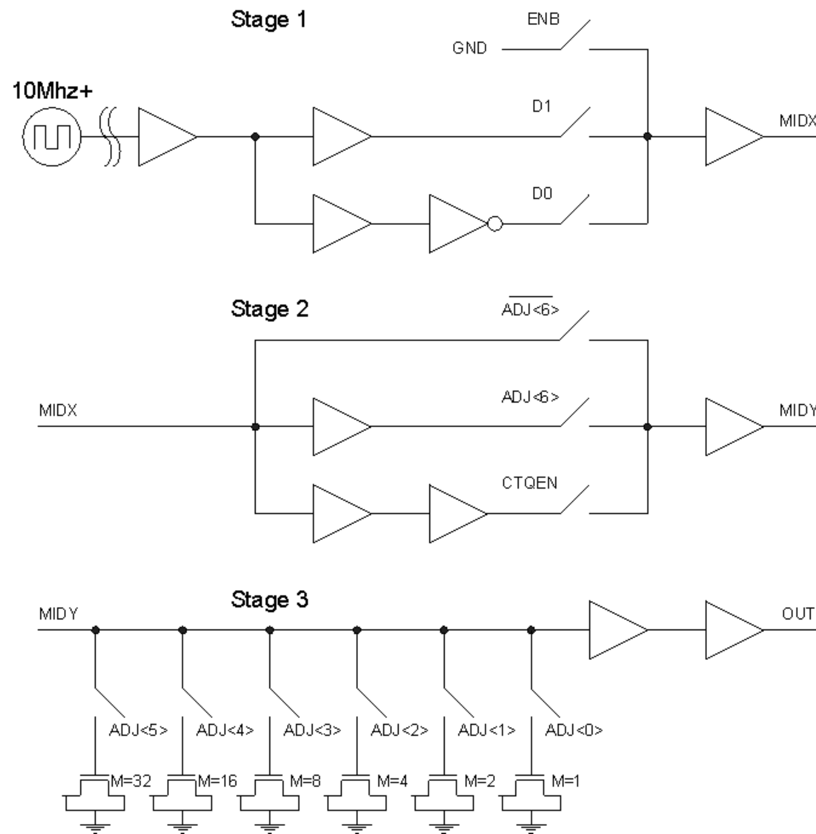


Figure 21 – Variable delay element schematic.

of delay at 1ps intervals. Since the combination of the minimum requirement (10ps) and continuous range are very difficult to meet using a single element, two identical delay elements are put in parallel, as shown in Figure 20, one set to a large, but minimum (with respect to the possible settings) delay and another with an adjustable delay. For measurements, the same signal will be sent through both delay elements and reach the flip-flop's inputs with a difference in arrival set by the adjustable skew.

Figure 21 shows a more detailed schematic of the variable delay element. The first stage of the delay element is used to choose whether or not to invert the signal. This will be used exclusively in the delay element used to drive the data inputs to the flip-flops. The second stage is used as a course adjust on the delay element and can be adjusted to

introduce roughly 30, 60, or 200 picoseconds of delay. The largest setting is to be used for the large setup time condition during a clock-to-output delay measurement where the other two settings are used to cover the possible range of setup times. The final stage is used as the fine adjust and introduces delays of roughly ½ picoseconds per adjust setting over a nominal range of 32 picoseconds. This is done by switching on up to 64 capacitors

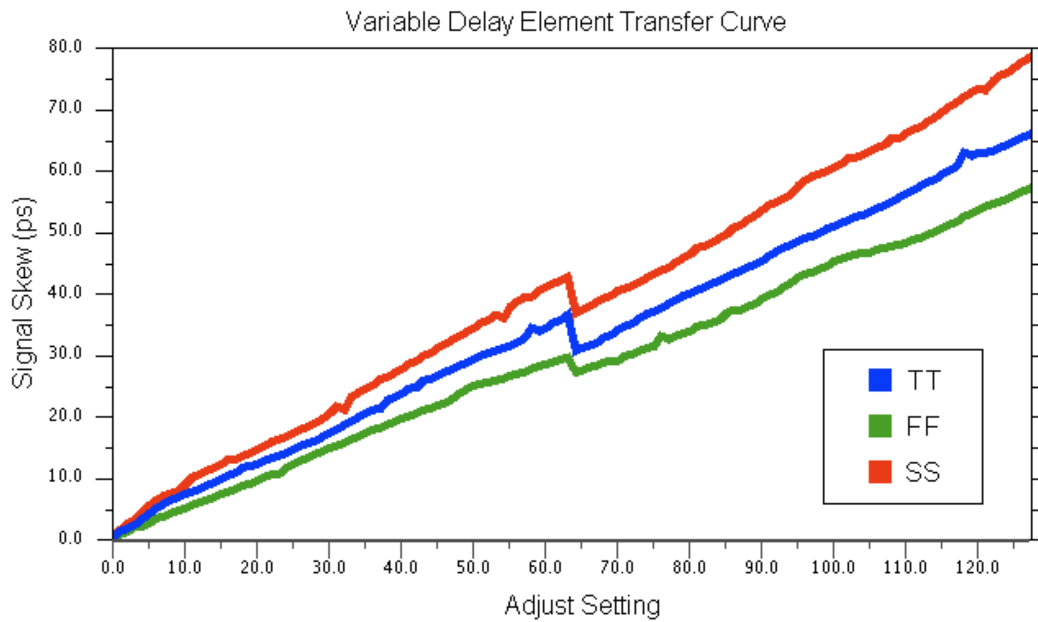


Figure 22. Digital-to-time transfer curves for variable delay element (TT, FF, and SS Corners).

at the load of a large inverter (so that the load change is roughly linear) and then subsequently regenerating the slope of the signal with a large buffer before outputting to the flip-flops. The simulated transfer curve for this digital-to-time converter for different corners is shown in Figure 22. The switch between course settings is intentionally skewed to introduce redundancy in delays so that there are no discontinuities in sweep value across all corners.

5.3 Timing Parameter Measurement

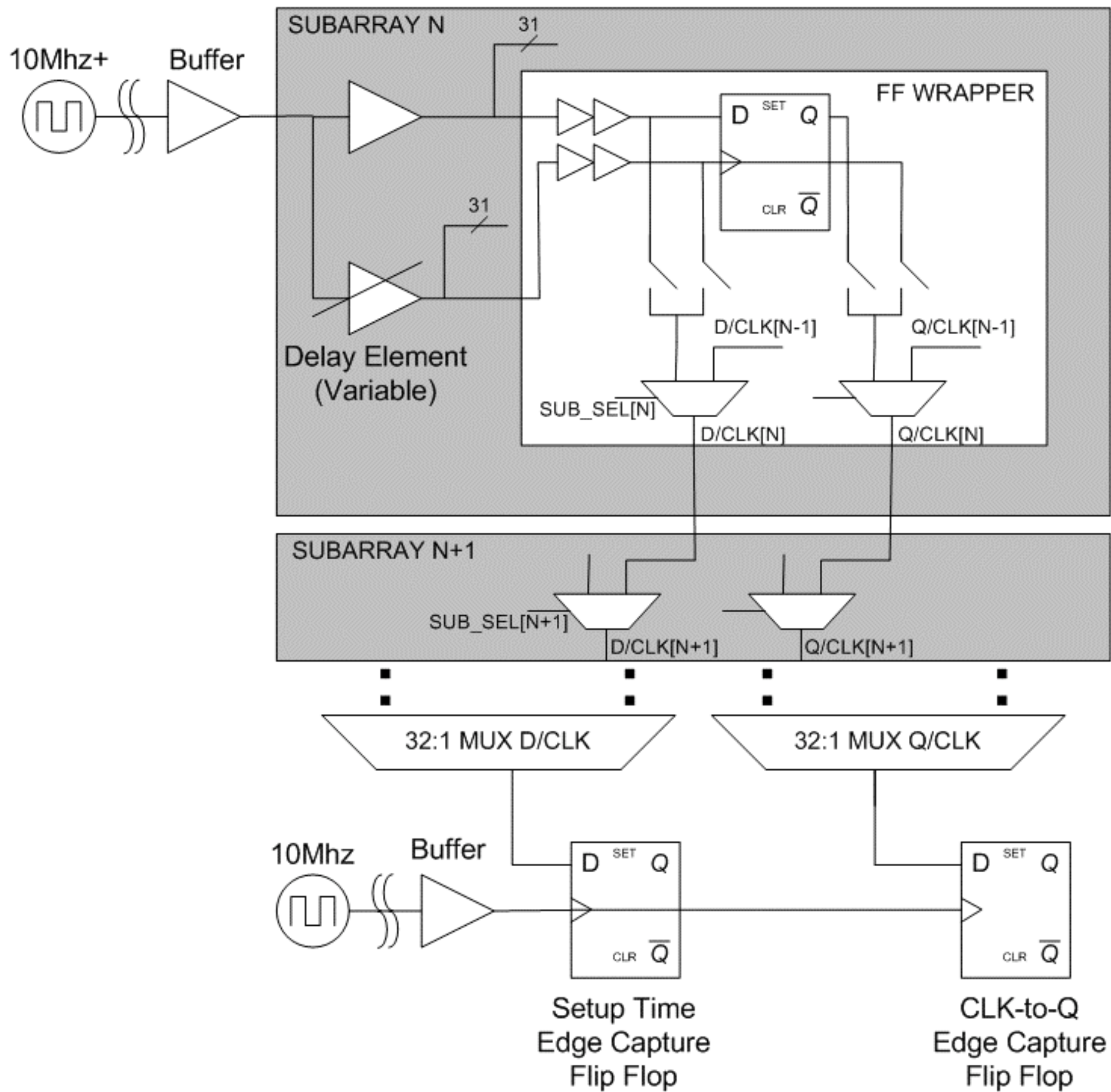


Figure 23. Absolute delay measurement circuit schematic.

One important source of variability in this circuit that needs to be accounted for is in the binary-weighted fine-adjust final stage of the delay element. If all of the variations from the expected delay of the five least significant bits reduce the actual delay through the delay element, and the last 32 unit elements associated with the most significant bit are all skewed such that they increase the delay from the expected value, there will be a

large discontinuity in the transfer curve. To account for this problem, we have two additional settings on the "minimum" delay element driving the clock that can skew the transfer curve to the left by approximately 2ps or 16ps. Thus, we can run the sweeps using these additional transfer curves and bin the results so that all possible signal skews are spaced out by the approximate resolution we desire.

5.3.1 Setup Time Measurement

In the final implementation of the chip, there are 32 pairs of delay elements each driving 32 flip-flops of the same test configuration. As shown in Figure 23, each flip-flop has a set of switches associated with it, as well as an output multiplexor. The setup time of each flip-flop is measured in three different steps. First, the skew is swept downwards from the largest setting that can be introduced between the clock and the data input, effectively ensuring that the setup time is met and the data, whether it is a '1' or '0', is captured correctly for the first few steps. Each skew step should be tested several times to average out the effects of supply noise and the region of uncertainty around the failure point of a flip-flop. The step value that results in 50% failures can then be recorded as the setup time setting for that flip-flop. During this mode, the switches that would normally be outputting the flip-flop's data and clock signal are turned off, and dummy switches that are loaded by dummy multiplexors are turned off. This is done so that the delay conditions when the data and clock signals are being measured and being output are the same.

In the output mode, a pair of large switches chooses between outputting the clock or data input of the flip-flop to a multiplexor. The multiplexor chooses whether to output the local signal or pass the previous sub-array's setup time signal through the rest of the

sub-array multiplexor chain and finally to the edge capture flip-flop used for setup time measurement. Again, because of asymmetries in the load of each flip-flop input depending on which is being sent out, dummy circuits are used to keep this difference from the original setup time characterization to a minimum.

5.3.2 Clock-to-Output Delay Measurement

Clock-to-output delay measurement of a flip-flop using the delay elements is done using the same methodology. Instead of setting the skew between the flip-flops to some characterized setup time, the largest skew that the delay elements can produce is used. As long as this skew is sufficiently high such that all the internal nodes of the master latch have settled, the variation between different delay element pairs will not significantly impact the reference-ability of this measurement scheme. A separate pair of matched switches is used to choose between the flip-flop clock and 'q' output since the setup time measurement circuit already requires the 'q' output to detect failures.

6. Conclusion

6.1 Summary

In this report, we have detailed the implementation of a series of test structures to study the variability of flip-flop timing parameters. First, we propose to study the variability of individual flip-flop building blocks by placing these inverting blocks into ring oscillators and measuring the average delay over several stages. While this measurement does not give us an exact delay value for any particular gate, it is still instructive to compare the relative variations (sigma/mean) of the various gates. Simulation results support our hypothesis that transistor stacks with a transistor switching at the output of the gate experience higher relative variation than other transistor stacks due to a larger percentage of its propagation delay spent in the more variable low-V_{GS}, high-V_{DS} region of operation.

Next, we attempt to form a more relevant comparison by creating eight different flip-flop configurations to study the effect of interchanging various building blocks at timing parameter-sensitive locations within a master-slave flip-flop. This allows us to isolate both the gate locations as well as the building blocks that affect the variability of timing parameters the most. We also propose that the metric to judge the variability tolerance of a flip-flop configuration's timing parameters should be the mean of the timing parameter plus 3σ, allowing us to account for the worst-case overhead a particular flip-flop configuration can present. When this metric is used to compare the flip-flop building blocks, we find that the previously discussed relative variation of a building block is not a relevant measure of its variability tolerance; in general, the transistor stacks with the least relative variation have a much larger delay than their counterparts (for example, the bottom-switching NMOS stack versus the top-switching NMOS stack) and thus the lower

variations from the larger delay still results in a gate with a larger overhead in delay. In other words, from the standpoint of minimizing the overhead of a flip-flop with fixed size and topology, the faster gates will still perform better even when accounting for variability. Thus, it is of interest to minimize the delays that correlate directly to the timing parameters. More specifically, for optimal setup time, the feedback inverter in the master latch should contain as much of the non-critical functionality as possible (set and reset implementation) so that the path critical to determining setup time can be implemented with as few stacks as possible. Again, simulation results confirmed our hypothesis. Finally, three different test structures are proposed that have enough resolution to measure the variability of flip-flop timing parameters.

6.2 Future Work

While the simulation results showed that variability is not as large of an issue with respect to flip-flop timing parameters, it is necessary to keep in mind that these results are based on an imprecise model. Actual process variations could potentially have a larger impact than what is modeled. A test chip was designed with implementations of the proposed test structures and actual measurements still need to be made to confirm the simulation expected results. If this experiment were repeated with less area constraints, different baselines for comparison could be used. For example, while we discussed earlier that keeping transistor area constant is a fair way to compare different topologies, it may also be interesting to optimize the nominal timing parameters of the various topologies and then compare the absolute variations. Furthermore, the scope of this study was limited to master-slave type flip-flops. As scaling further increases the impact of variability, if the

proposed test structures, in particular the absolute delay measurement circuit, can be shown to achieve their theoretical resolutions, the same structures can be used or modified in order to explore variability in more complex flip-flop topologies as well as other digital building blocks.

References

[1] L.T. Pang, B. Nikolić, "Impact of layout on 90nm process parameter fluctuations," Symposium on VLSI Circuits, 2006.

[2] L.T. Pang, "Circuits for measurement and analysis of CMOS variability," doctoral dissertation, Dept. Electrical Engineering, Univ. of California, Berkeley, 2008.

[3] Z. Guo et al., "Large-Scale Read/Write Margin Measurement in 45nm CMOS SRAM Arrays," to be presented in Symposium on VLSI Circuits, 2008.

[4] M. Khella et al., "Wordline & Bitline Pulsing Schemes for Improving SRAM Cell Stability in Low-Vcc 65nm CMOS Designs," Symposium on VLSI Circuits, 2006.

[5] H. Pilo et al., "An SRAM Design in 65nm Technology Node Featuring Read and Write-Assist Circuits to Expand Operation Voltage," *IEEE J. Solid-State Circuits*, vol.42, Apr. 2007.

[6] V. Kheterpal et al., "Design Methodology for IC Manufacturability Based on Regular Logic-Bricks," Design Automation Conference, 2005.

[7] A. Hastings, *The Art of Analog Layout, Prentice Hall*, 2001.

[8] L. Pileggi, "Variation-Tolerant Analog and Digital Design Methodologies," ISSCC 2008, Microprocessor Forum.

[9] V. Stojanovic, V. G. Oklobdzija, "Comparative analysis of masterslave latches and flip-flops for high-performance and low-power systems," *IEEE J. Solid-State Circuits*, vol. 34, Apr. 1999.

[10] M. Abas et al., "Design of Sub-10-Picoseconds On-Chip Time Measurement Circuit," Design, Automation and Test in Europe Conference and Exhibition, 2004.

[11] N. Abaskharoun et al., "Strategies for On-Chip Sub-Nanosecond Signal Capture and Timing Measurements," IEEE International Symposium on Circuits and Systems, 2001.

[12] N. Nedovic et al., "A Test Circuit for Measurement of Clocked Storage Element Characterisitics," *IEEE J. Solid-State Circuits*, vol. 39, Aug. 2004.