Bachelor's thesis

Degree programme of Electronics

Electronics

2015

Toni Nylund

DEGRADATION OF INTEGRATED CIRCUITS DUE TO SCALING IN FPGAS

TURUN AMMATTIKORKEAKOULU
TURKU UNIVERSITY OF APPLIED SCIENCES

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Electronics | Electronics

2015 | 26

Instructor: Timo Tolmunen

Toni Nylund

DEGRADATION OF INTEGRATED CIRCUITS DUE TO SCALING IN FPGAS

Decreased size and improved performance in integrated circuits have brought up a number of problems that are directly related to the decreasing life cycle and reliability of modern microchips. These negative effects can be studied with a Field Programmable Gate Array (FPGA) by measuring the internal delays of known programmable logic elements on the die.

This thesis consists of the introduction to the problem through some of the main components of Integrated Circuit (IC) degradation due to scaling, a presentation of FPGA and its capabilities concerning this subject and an analysis of different possibilities to identify the problem by an on-chip FPGA testing design.

As a result, an easy-to-understand but thorough document was created, which gives a strong basic understanding of the different factors of IC scaling from the point of view of a FPGA designer. The purpose of this thesis is to give information that will help to resolve the basic misunderstandings of the technology and to enable a person to understand the subject without extensive digital hardware design knowledge. The results illustrate the inner workings of a fast microchip design and how the failures in operation are caused.

KEYWORDS:

scaling, integrated circuit, FPGA, degradation, reliability, delay measurement

OPINNÄYTETYÖ (AMK) | TIIVISTELMÄ

TURUN AMMATTIKORKEAKOULU

Elektronikka | Elektroniikkasuunnittelu

2015 | 26

Ohjaaja: Timo Tolmunen

Toni Nylund

MIKROPIIRIEN RAPPEUTUMINEN FPGA-PIIREISSÄ

Mikropiirien valmistusteknologioiden kehityssuunta aina vain pienempiin komponenttikokoihin ja parantuneeseen suorituskykyyn on johtanut mikropiirien elinkaarien lyhentymiseen. Tätä ilmiötä on oivallista tutkia Field Programmable Gate Array (FPGA) -piirillä sen nykyaikaistuneiden kellogeneraatio-ominaisuuksien vuoksi. Mittaamalla tunnettujen logiikkaelementtien viiveitä sirun eri kohdissa voidaan määrittää sirun rappeutumisen taso sekä sen herkimmät kohdat.

Työssä selvitetään suurimmat rappeutumiseen vaikuttavat tekijät, jotka ovat suoraa seurausta pienentyneestä komponentti- ja sirukoosta. FPGA-piirien käyttö ja soveltuvuus aiheeseen perustellaan, ja toimintaa havainnollistetaan. Erilaisten mittausteknologioiden ominaisuuksia tarkastellaan ja lopuksi verrataan.

Tuloksena luotiin useisiin alan tutkimuksiin perustuva, perus- ja korkeatasoisen tiedon yhdistävä ja tiivistävä paketti. Työssä myös perusteltiin nykyaikaisille FPGA-piireille sopiva ja tehokas mittausteknologia, ja syyt miksi perinteinen teknologia ei ole välttämättä käyttökelpoista. Tulos auttaa havainnoillistamaan nopeiden mikropiirien toimintaa ja miten virheet niissä syntyvät.

ASIASANAT:

mikropiiri, integroitu piiri, pienentyminen, komponenttikoko, FPGA, viive, väylä, mittaus, luotettavuus

CONTENT

PICTURES, FIGURES

LIST OF ABBREVIATIONS

1 INTRODUCTION 1

2 INTEGRATED CIRCUIT MANUFACTURING 2

3 THE NEGATIVE EFFECTS OF THE SHRINKING DIE SIZE 4

3.1 Temperature 5

3.2 Electromigration 6

3.3 Dielectric Breakdown 8

3.4 Process Variation 9

4 FPGA IN RELIABILITY RESEARCH 10

4.1 The Reliability Issue in FPGAs 11

4.2 Reliability Evaluation in FPGAs 12

5 RING OSCILLATOR DESIGN 14

6 OFFLINE FREQUENCY-SWEEP DESIGN 16

7 ONLINE FREQUENCY-SWEEP DESIGN 20

8 CONCLUSIONS 24

REFERENCES 25

PICTURES

Picture 1: Comparison between Intel's 32 nm and 22 nm manufacturing processes [6] 3
Picture 2: A damaged interconnect with visible voids and disintegration [12] 7

FIGURES

Figure 1: Charts of technology size against peak temperature and MTTF [10] 5
Figure 2: Block design of a ring oscillator with a capture counter [20] 14
Figure 3: Design of a frequency-sweep-based delay measurement circuit [17] 16
Figure 4: Timing diagram of the offline measurement over three cycles [17] 17
Figure 5: Charted results from the offline measurement [17] 18
Figure 6: An online phase-sweep-based delay measurement design [19] 20
Figure 7: Timing diagrams of passing and failing online measurements [19] 21
Figure 8: Charted results after 360° phase sweep on the test clock [19] 22

LIST OF ABBREVIATIONS

Abbreviation	Explanation of abbreviation (Source)
ASIC	Application-Specific Integrated Circuit
CUT	Circuit Under Test
EDC	Error Detection Circuit
FPGA	Field Programmable Gate Array
IC	Integrated Circuit
MTTF	Mean Time To Failure
PLL	Phase-Locked Loop
PUM	Paths Under Measure

1 INTRODUCTION

For several decades integrated circuit (IC) manufacturers have been aiming to develop their manufacturing processes into creating ever smaller feature sizes on silicon wafers to improve the performance and to shrink the size of their chips. The increased performance and decreased size, however, comes with a cost of the chips being more sensitive to variances from manufacturing and reduced long-term durability. Field Programmable Gate Arrays (FPGA) as reprogrammable semiconductor devices have become a victim for these effects, but also risen up as great tools to research this problem.

The purpose of this thesis is to study the subject of IC reliability and gather information from multiple research papers with practical examples of FPGA designs. The paper starts from presenting some basics of IC manufacturing and the main negative consequences of scaling of ICs to introduce the problem. FPGAs are introduced as tools to research the issue. Analysis of the mechanics of reliability testing on FPGAs is presented with the help of three example methods. The examples are evaluated and presented by their properties. The core idea of this thesis is to create a document that can quickly walk through the basic and the advanced material to help understanding the subject.

The concept of reliability testing in FPGAs is quite well researched and papers of different designs and techniques can be found easily. The problem in many of these papers is that they contain only very advanced form of information that can be very hard to understand without extensive knowledge of digital hardware design. As FPGAs and their development tools have become more capable and easier to use, extensive knowledge is not required to use them to create designs anymore. Therefore documents that are easier to understand should be created to ease the spread of FPGA designing.

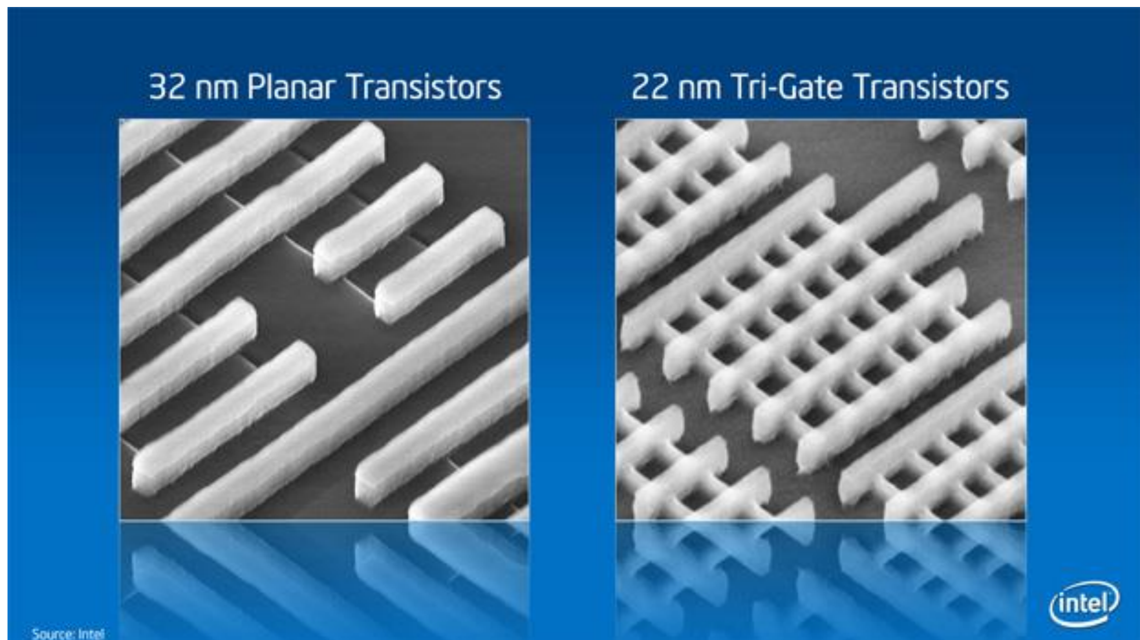
This thesis will focus on the FPGA designs for the reason that they give great practical insight to the inner workings of an IC and that the results are easy to interpret.

2 INTEGRATED CIRCUIT MANUFACTURING

Integrated circuit manufacturing has evolved immensely from the days ICs were first discovered in 1959. The first produced ICs had around a dozen components each whereas currently a normal PC has a processor chip that is not only very small but also likely contains more than a billion components. [1, 2] Pictures of old computers that take up enough space to fill a small room are just something to laugh at for the people of today who are carrying modern smart devices with them on a daily basis. Among other performance adding technologies, the scaling of the manufacturing process size has pushed the capabilities of ICs to a whole another level from before and enabled the information technology-filled world that we live in today.

Even though the ICs today are cheaper than ever, the manufacturing costs have gone up with the more advanced technologies that are required. The sudden explosion of demand for ICs in the form of intelligent devices has managed to keep the manufacturing process profitable but if the manufacturers want to be able to keep up with Moore’s Law in the future they need to create better and more cost-efficient technologies. [4]

A likely new technology for the future is the nanotechnology which would take the manufacturing process to the molecular level. [5] It could be able to keep the Moore’s Law still going for a while, perhaps all the way until the physical limitations allow no more shrinking. One can hope that the future brings manufacturing methods that not only enable the even more shrinking components but also manages to keep the manufacturing costs tolerable. It should be acknowledged that the costs are not the only problem in creating smaller and denser ICs.



Picture 1: Comparison between Intel's 32 nm and 22 nm manufacturing processes [6]

The manufacturing process is named after the feature size that it is capable to do. The process sizes have gone down from 10 µm (1971) to 1 µm (1985), leading to 180 nm (2000) and eventually 22 nm (2012). [3] Picture 1 illustrates the advancements that a smaller feature size can achieve. The feature size refers to the smallest possible feature that can be printed on the wafer. It does not refer to the component size itself but the smallest possible features that the component can be manufactured with. Smaller feature sizes enable more components on the same space or the ability to shrink the die without losing performance.

As the feature sizes get smaller the tolerance for defects also get reduced. This is taken to account in the manufacturing process to maximize the yield of functioning chips. However, the tolerance for post-production defects remains reduced. The logic sensitivity has increased massively with the shrinkage of die sizes even though its effects can and has been made less significant by system design, especially with newer process technologies. [7]

3 THE NEGATIVE EFFECTS OF THE SHRINKING DIE SIZE

With the continuity of tolerance reduction for post-production defects in new manufacturing processes, the causes for these negative effects need to be more thoroughly researched. The manufacturing industry has huge pressure to produce ICs fast and cheap to be able to survive in the markets and long-term reliability is often neglected [8]. Knowing the mechanisms that are related to the problems caused by the smaller component and die size helps to understand the reliability of ICs better and how it may change in the future.

This section introduces the three main negative side effects that are caused by the more component-packed and denser ICs. A direct relation in this context is the negative effect being caused by the physical decreases of the feature size and die area.

In short, the negative effects that are directly related to the shrinking die size are increased temperatures in the hottest structures on the die, the decreased tolerance to the electromigration-caused degradation of interconnects and the increased vulnerability to the effects of dielectric breakdown on the gate oxides. [9] The concept of process variation is also introduced as it is a problem that has become more severe with the scaling of ICs, even though it is arguable if it is directly related to the scaling itself. Other indirect negative effects are mostly related to the increased temperatures and will be treated as an effect caused by temperature and not scaling.

3.1 Temperature

Temperature problems are very well known in the electronics industry. Getting devices to work under different specifications, loads and environments has required thermal planning from the designers for a long time now. The scaling of the feature sizes can cause problems in terms of risen temperatures, but the overall temperatures of ICs have not changed with scaling. The core of the problem is in the temperatures of the hottest structures, which affects the peak temperatures of the chip. This can be seen on Figure 1.

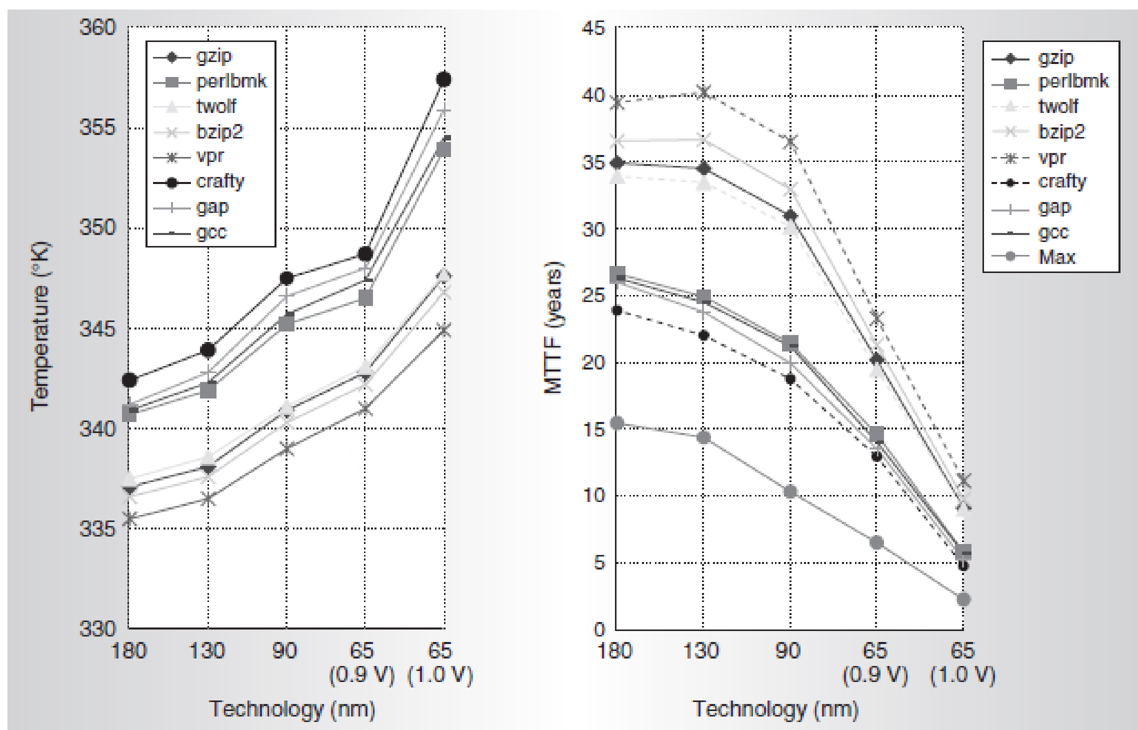


Figure 1: Charts of technology size against peak temperature and MTTF [10]

With the shrinking die size the number of components on the die increases while their distance from each other decreases, creating denser chips. Inside the chip the power density increases as well as the power leakage which results in the higher overall peak temperatures, which are caused by the increased temperatures in the hottest structures. The amount of power leakage has an exponential relation with temperature, which leads to quickly worsening conditions in high temperatures. [9]

High temperature is directly related to the severity of other damaging reactions such as electromigration, stress migration and dielectric breakdown. It behaves either as an accelerator or as an enabler for these negative effects. Some of the phenomena have an exponential relation with temperature, making the rising of peak temperatures the most influential negative side-effect of scaling for the reliability of produced ICs.

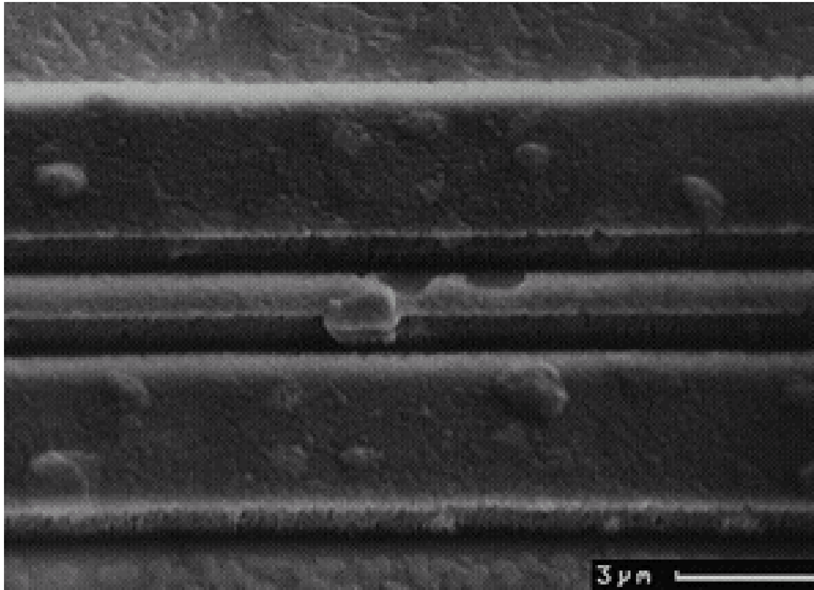
For example, stress migration has not increased directly because of the scaling but the increased temperatures tend to accelerate this phenomenon, making hotter chips more vulnerable to stress damage over time.

As the scaling of manufacturing processes continues, the temperatures of the hottest structures can be expected to grow to be a bigger problem and will decrease the long-term reliability of the produced chips by indirect causality. [9, 10]

3.2 Electromigration

Disintegration of the metallic conductors, or interconnects, creates an alarming threat to the reliability of a chip. The disintegration is caused by the momentum transfer between electrons and the ions that make up the interconnect material. This disintegrative reaction is called electromigration and it has become one of the main points of focus when battling the causes for reduced reliability in denser designs. [11, 13]

The scaling of the components on a chip also reduces the amount of the interconnect-material, creating an increased vulnerability to the effects of electromigration. In the worst-case scenario electromigration might lead to the loss of connections entirely. Though the total loss of connections is rare, the damaged conductors will have a negative effect on the reliability of the chip and the risk of total chip failure will increase as the feature sizes keep getting smaller. [9, 10]



Picture 2: A damaged interconnect with visible voids and disintegration [12]

The mechanism is usually enabled by an indirect effect such as a temperature spike, which with the rise of the current density in a conductor creates a void on to the conductor. This void then enables a loop-like reaction where it creates permanent increase in the current density at around itself, creating more heat that leads to even greater current densities and to the increase of the size of the void. You can see this type of damage on a interconnect in Picture 2. [11, 13]

Another effect of the disintegration of metal conductors is the increase of the electric resistance of the material. As the conductor changes its shape and density, the resistive properties also change. Because the reaction is disintegrative towards the material, the resistivity of the material is assumed to increase over time, affecting the overall reliability. [9, 10]

3.3 Dielectric Breakdown

Dielectric breakdown refers to the losing of material insulating properties. Parts that should be insulators begin to conduct electricity causing unreliability. Gate oxide refers to the dielectric layer manufactured into MOSFETs to separate the internal source and drain terminals from the gate. The thickness of the gate oxide is decreasing along with the scaling, making the chip more vulnerable to the effects of dielectric breakdown. [14]

Gate leakage in MOSFETs is a phenomenon where leakage currents travel through insulators. Dielectric breakdown creates an increase in gate leakage in MOSFETs. The leakage increases exponentially as the gate oxide becomes thinner. As the already thin oxide layer starts to wear out due to aging, the long-term reliability of the transistor is compromised. Single transistors become more vulnerable to failures at the same time as the transistor counts in ICs keep increasing, creating more unreliability in the produced ICs. Temperature also affects the gate leakage and tunneling currents, which increases the probability of failure in denser and hotter chips. [10]

The tunneling current is a quantum phenomenon where electrons can jump through barriers and create current. As the barrier, gate oxide in this case, becomes thinner, the current increases. Though this phenomenon is not directly related to dielectric breakdown, its effects are the same; increased leak currents when the oxide layer thins. [15]

The main negative effects of gate leakage are increased power consumption and possibly even malfunction of the transistor. The performance of ICs is largely being held back by these effects and forces the manufacturers to compromise between performance, power consumption and reliability. [9, 10]

3.4 Process Variation

The previously mentioned direct negative effects of scaling will continue to exist even if they were simulated in an environment where every component is exactly as specified. In the real world there is always a slight variation in every component manufactured in terms of size, density or electrical properties. The tolerance for such variations will have to decrease along with the size of components to assure the functionality according to the desired specifications. Process variation is not directly caused by scaling but unless a zero-tolerance manufacturing technique is invented, it will grow to be a bigger problem just like the directly related issues. [16]

Process variations can be either differences in component dimensions or material purities. Mismatching dimensions can be thought as conductors or insulators having different lengths, heights or widths, creating different properties for the platform in different areas of the chip. Material impurities will affect the base electrical properties of the components, which will then be scaled by the dimensions of the applied material. Together these two variations can create a scenario where some components have the minimum acceptable dimensions while also having a high percentage of impurities in the materials, creating a component that does not have the same properties as a component on the other side of the tolerance spectrum. [16]

As the properties and dimensions of components differ from each other, the degradation of the chip will be uneven. In serial processing the reliability of the weakest components brings down the reliability of the whole system. The previously mentioned direct negative effects will cause more damage if the component is already weaker due to process variations. For example, if the gate oxide thickness and purity is varied between two transistors, dielectric breakdown will affect one of them more severely than the other.

4 FPGA IN RELIABILITY RESEARCH

FPGA is a semiconductor device that's advantages lie in its flexibility to change its functionality. The field-programmability refers to the ability to program the functionality of the device after it has been manufactured, and it can be reprogrammed for updated or different designs. Its counterpart is ASIC (Application-Specific Integrated Circuit) which is manufactured for a specific function that cannot be changed afterwards. In the past years the capabilities of FPGAs and design tools have developed to a point where they can feasibly be used without extensive knowledge of digital hardware design, which has increased the use of FPGAs tremendously. [17]

Like other semiconductor devices, FPGAs have benefitted greatly from the scaling of manufacturing processes. The number of components in a FPGA chip not only improves the performance and power consumption like in ASICs, but also expands the overall capabilities for more complex designs that can be implemented on to the FPGA. On the contrary, the dependence on meeting the timing requirements of the signals through the design makes the reliability question especially relevant in FPGAs to assure that the faster and more complex designs function as expected.

ASICs and FPGAs have a big difference on how the timing of signals is taken into account in the design process. In ASICs a whole design is required before manufacturing and configuring the device, which means all the most critical paths of the design are known from the beginning and the device can be built with this information in mind. FPGAs, however, are not configured to any specific design until a designer decides to do so, making the timing analysis a much more tedious and time-demanding process for the designer. Instead of building the hardware for the desired application, the FPGA designer needs to accommodate the design on to the existing FPGA hardware. [18]

4.1 The Reliability Issue in FPGAs

FPGAs have become exceptional tools for researching the actual effects of process variation and aging in ICs for two reasons; the FPGAs themselves have become more vulnerable to these factors and the capabilities of FPGAs have developed to a point where an accurate self-test circuitry can be programmed and run on the device. To evaluate the condition of the device and to research the practical behavior is beneficial for the designers as well as the researchers. [18]

FPGAs are now used in various different industries and the demand for reliable and predictable functionality has increased. The reliability issues can cause complications of different magnitudes in different industries and a faulty FPGA could, for example, compromise safety. This has been globally acknowledged and studies of different techniques to evaluate the reliability of an FPGA have emerged. [19]

Uneven degradation of the chip causes unpredictability in an FPGA. FPGA can be said to be parallel in its nature, which means the reliability will vary depending on the area. The parallel nature creates huge advantages for FPGA versus serial processing circuits in certain operations, but it can also create unpredictability into the functionality. For example, if the parallel operations run on logic fabric with different properties, the operations will run with unpredictable speed differences. In serial processing the slowest path will designate the performance. As the designer is responsible for routing the design through the logic fabric, he should know which areas of the chip have possibly been degraded to avoid having the most timing-sensitive paths on the slowest areas. [18]

4.2 Reliability Evaluation in FPGAs

As mentioned before, currently you can find various techniques to accomplish some level of device integrity verification on FPGAs. These techniques can be separated into online and offline methods, depending on if, for example, the analysis is wanted to perform for the whole chip or just for some of the specific critical paths on a design. A use case for an online measurement would be to constantly monitor the degradation level of a critical path on a design to foresee a possible impending failure. Offline measurement could be done to map the degradation level throughout the whole chip area to give the designer an idea of the usability of the chip before implementing any designs on it. [18, 20]

Whether an online or offline method is used, the basic principle of evaluating the reliability and consistency of functionality of an FPGA is to measure the delays of paths on a circuit. The delay values through different paths and logic elements give a reference point of the condition of those specific areas. Ideally, the values should be same throughout the chip but by mapping the whole chip it can be seen that the process variations and uneven degradation gradually makes the chip to have different delay values in different areas. The delay value of a specific path can also be measured over time to detect the possible degradation and its speed. [18, 20, 21]

The main points of a good measurement method are high accuracy, fast measurement and low resource requirements. The reason for the reliability research on FPGAs being a fairly recent discovery is because early FPGAs could not facilitate a design that would have passed the requirements of the aforementioned main points. The current generations of FPGAs are equipped with advanced clock generators that allow these requirements to be met without external hardware. [18, 20]

In the next section of this thesis a few existing on-chip reliability evaluation techniques are introduced to give a better idea of the process, and to give deeper understanding of the delays in FPGAs and other ICs. The great advantage of FPGAs is the ability to implement an on-chip delay measurement design without extra costs, therefore all the following techniques are considered to be on-chip solutions and capabilities of possible external measurement circuits made with same principles are not taken into account when evaluating a method's properties in delay measurement.

5 RING OSCILLATOR METHOD

One of the traditionally most used methods for circuit delay measurement is done by implementing a ring oscillator design and observing its output frequency in different conditions. This has been used extensively for different ICs and with the rise of popularity of FPGAs it was adopted as one of the early performance and reliability estimation techniques for FPGA chips. While this method gives some information of the condition of the FPGA, an on-chip self-test implementation lacks accuracy, consistency of operation and feasibility of implementation to be the best choice for modern FPGAs. [18, 21]

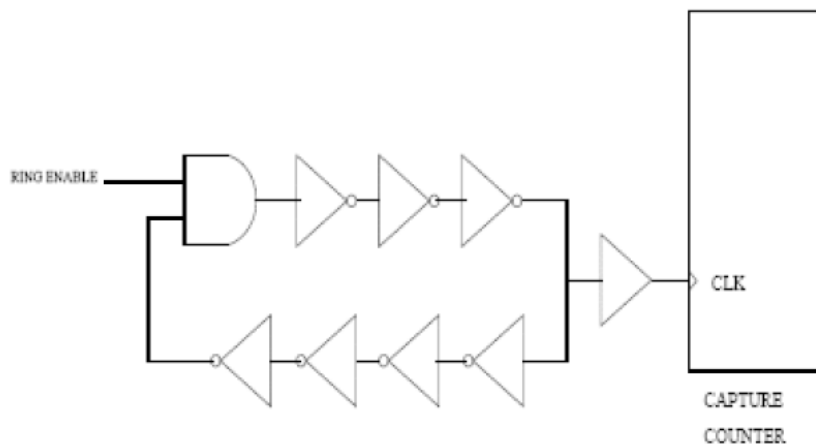


Figure 2: Block design of a ring oscillator with a capture counter [21]

Figure 2 shows the basic design of a ring oscillator with a capture counter. Implementing a ring oscillator on a FPGA might seem like a viable choice but as the design needs to be implemented on to the same fabric as the tested circuit, the first impacts on accuracy are already present in the hardware. Each inverter in the oscillator adds its own delay in to the design and, in theory, by knowing the number of inverters and the output frequency, a delay measurement can be accomplished. However, this measured delay is acquired from the sum of all the delay-incrementing logic on the oscillator and the information can be misleading due to the variations inside the oscillator. As the number of inverters increases, so does the variability and inconsistency. [18]

The minimum delay between two logic elements is the same as the maximum clock frequency that can be travel through them. The ring oscillator design can be made tunable and a frequency sweep can be applied to find the frequency that is too fast for the path. With FPGAs this method is not feasible since the absolute frequency of the output and the impact of the internal delays of the oscillators cannot be determined without extensive monitoring effort. [18]

Ring oscillators are highly sensitive to external factors such as temperature and feeding voltage. To create an accurate ring oscillator the user needs to keep all external factors from fluctuating. Modern dense chips with possible variations in temperatures in different structures might make the operation of a ring oscillator inconsistent. Therefore, running the design to capture the delay map of a whole chip could show incorrect data for the densest parts.

In an FPGA implementation the designer needs to place and route the design. To create an accurate ring oscillator, the placing should be done within uniform distances between the used inverters. This will require extra effort from the designer, and to make sure that the automated design tools do not misplace any viable parts of the design, the tools either cannot be used or specific parameters need to be added.

Since the frequency provided by a ring oscillator in an FPGA fluctuates, the data needs to be gathered from a time frame where the fluctuation is at minimum. The test run needs to be long enough for such a time frame to come up and as it is very random, the testing time cannot be accurately specified. Automation for detecting such time frames will require more logic added into the circuit or effort from the operator to find them by hand. [21]

In summary, even though ring oscillators are very widely used in delay measurement operations in various different stages from manufacturing to later degradation studies of ICs, it is not a feasible choice in an on-chip FPGA implementation. Currently available documents of delay measurement in ICs are largely circled around the ring oscillator approach, but for modern FPGAs there are better methods.

6 OFFLINE FREQUENCY-SWEEP METHOD

The reference logic of the design used in this work is presented in [18]. The idea is to use the on-chip capabilities of an FPGA to create a built-in self-test (BIST) circuitry that offers accurate results with small resource requirements.

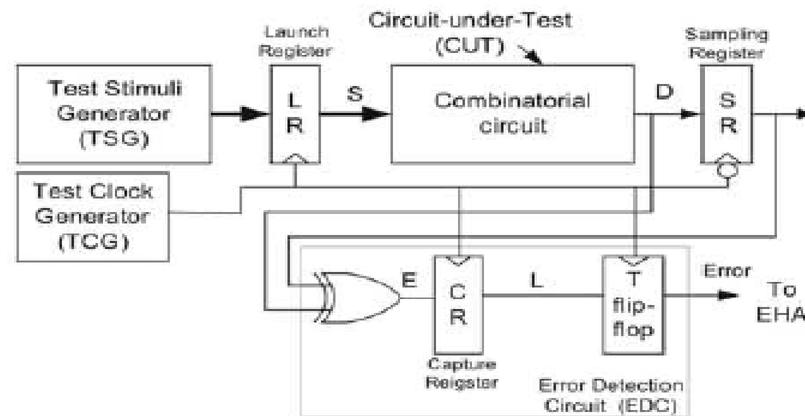


Figure 3: Design of a frequency-sweep-based delay measurement circuit [18]

The flexible on-chip clock generators in modern FPGAs combined with a simple error detection circuit (EDC) can be used to accurately estimate the effective delay present in any logic element or path. Figure 3 illustrates this design. The circuit under test (CUT) is placed between two pipeline registers that are clocked by the scan clock at the opposite phases. Increasing the scan frequency means the delay of the CUT will have a larger influence on the operation. Once a certain threshold frequency is passed, the delay of the CUT will start to cause failures in the operation and the EDC will store the error rate. This threshold frequency can be used to estimate the delay of the CUT. [18]

The threshold point can be found by sweeping the scan frequency over the threshold area and monitoring the EDC output. The step size in the frequency sweep has a big influence on the accuracy of the measurement. With this design, recent generations of FPGAs can manage a timing resolution of 1 ps and lower, which is excellent considering the delays are most likely measured in the nanosecond scale. Also as the clock generators on FPGAs move towards

faster speeds, the timing resolution of this design will increase accordingly, making it suitable in the future as well. [18]

While the EDC output in this design already gives readable results, the operation needs to be analyzed more closely to understand them.

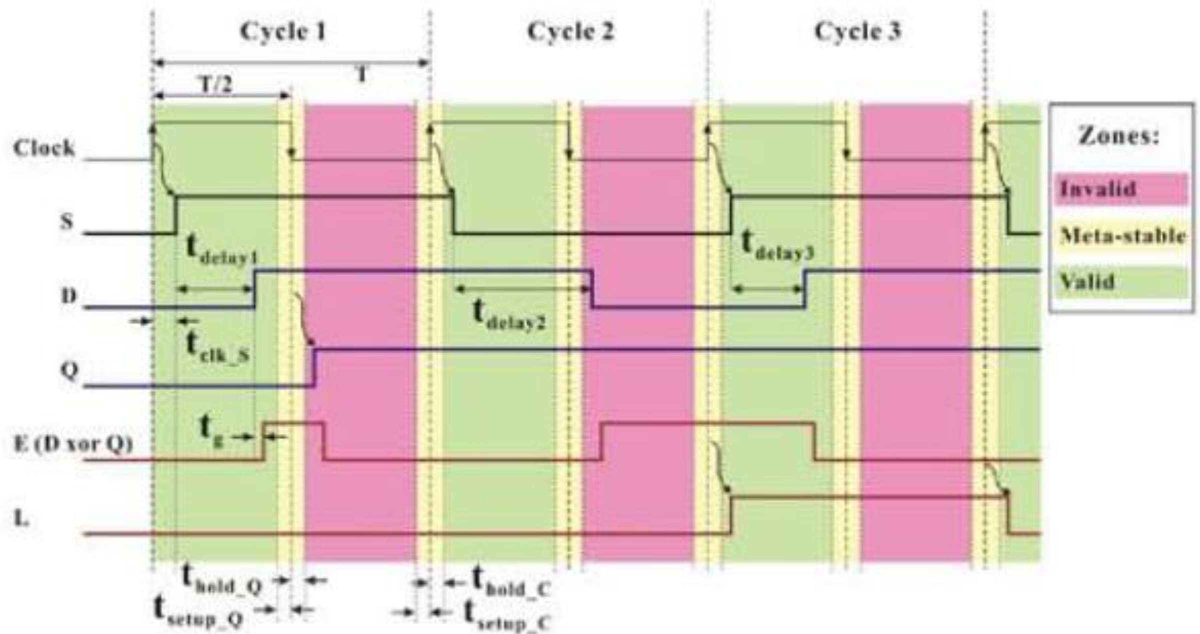


Figure 4: Timing diagram of the offline measurement over three cycles [18]

As seen from the differences between t_{delay1}, t_{delay2} and t_{delay3} on Figure 4, there is a variation between the delays of positive and negative transitions in the CUT. Also, the registers have their small setup and hold time requirements and the XOR gate has its propagation delay that should be taken into account [18]. The failure in operation can be seen in Cycle 2 when the value of D fails to transfer during the valid period and causes the output of Q to not transfer correctly. The EDC reacts to the failure of operation and after one clock cycle creates a high signal for a flip flop to register the error.

Failure rate of the CUT will increase with the scan frequency. The failure rate data can be used to build a chart that shows the percentage of failures against the scan frequency. Because of some fluctuations in the form of clock jitter, differences in the positive and negative transition delays and heating, the results will have some fluctuations too. Therefore it’s more feasible to run the

design multiple times and to gather all the data into a chart where the information can be visualized for better accuracy. [18]

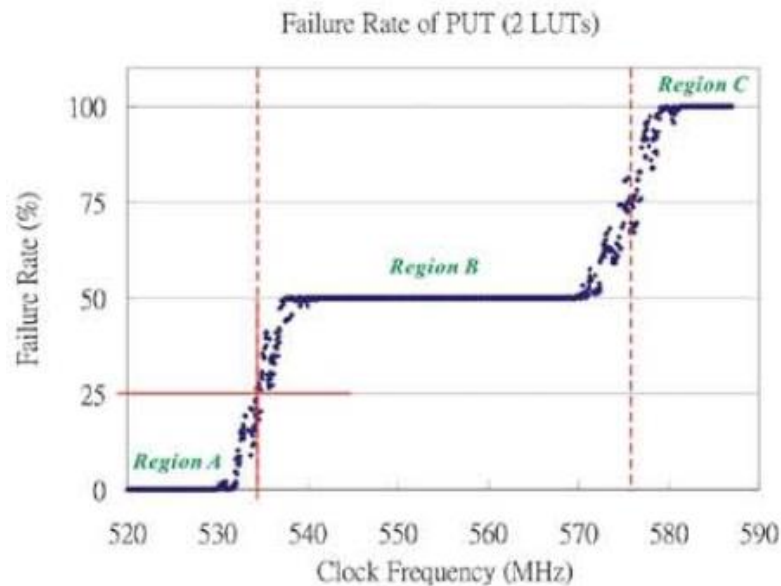


Figure 5: Charted results from the offline measurement [18]

Results gathered in a chart in Figure 5 shows some interesting aspects that will give more information about the operation. In Region B the failure rate seems to settle around 50%, which can be explained by the differences in delays of positive and negative transitions in the CUT. At a certain frequency range only one transition will fail while the other will transfer successfully, in which case the slower of the transitions is considered to be causing the failures. [18]

The delay of the CUT is not the only cause for the failures. Due to meta-stable periods in the operation and the jitter in the clock signal, some of the errors are dependent on the used FPGA. These factors need to be considered accordingly to the used device. In this example the delay of the slower transition is estimated to have a 50% effect on the failure and therefore the 25% spot is marked to give an estimation of the effective delay of the CUT. The estimation needs to be done with information of the FPGAs capabilities.

Size of the steps in the frequency sweep should be analyzed as well to see if there is variation. The possible variation is dependent on the clock generator of the FPGA and it can have a considerable effect on the accuracy. Newer generations of FPGAs are expected to perform better in this. [18]

While the design is simple in its basic construct and operation, the implementation is challenging when it is implemented in a parallel configuration to run tests on large chip areas concurrently. The parallel implementation requires automation for the control unit and data storage, as well as calibration of the value of the failure rate which determines the estimation of the CUT delay. The accuracy data from the tests of singular or small groups of logic elements can be used in the automated delay estimation process. After implementation the parallel design can be used for fast and accurate mapping of the variations on the chip.

Considering the clock jitter, meta-stability of the data transfers, temperature fluctuation in the power-up phase of the FPGA and other external factors, the design is suggested to be run multiple times to find the average values for the more accurate data charts. This creates more work in the implementation stage, but after the initial testing on a specific device, the gathered accuracy information can be used in the automation of the parallel implementation for long-term testing.

As this design creates a failure of operation in the CUT by a frequency-sweep, it cannot be used to concurrently test a running circuit. Therefore it is not usable to monitor a device that needs to be running a design uninterrupted at all times. This offline method can be used in manufacturing for self-test data or in evaluation of used FPGAs. It achieves better accuracy and usability than a ring oscillator -based design by taking advantage of the modern resources on the FPGA and by decreasing the testable area. [18]

7 ONLINE PHASE-SWEEP METHOD

Frequency-sweep-based methods require the system clock to change its frequency, forcing the tests to be made offline. Another design for an on-chip online delay measurement circuit is presented in [20]. It attempts to achieve online measuring capability by a phase-sweep and a function-mimicking shadow register.

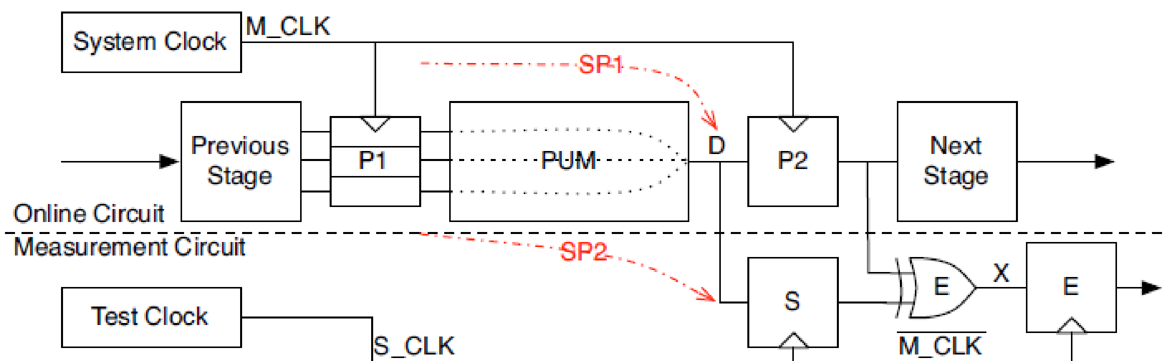


Figure 6: An online phase-sweep-based delay measurement design [20]

Phase-sweeping creates an interesting opportunity to mimic a part of a design and run it at varied time-differences in respect to the system clock. This enables the user to measure the available slack time in Paths Under Measure (PUM). Figure 6 shows the design. The slack time is expected to decrease over time due to degradation of the FPGA, and by monitoring it the user could foresee the possible impending failure. The slack time is measured with a shadow register, which gets the same input as the register it is mimicking in the tested circuit.

The shadow register is equivalent to the register it is shadowing. It is clocked at the same frequency as the system clock but with configurable phase. By sweeping the phase of the clock, the shadow register can be made to simulate worse timing-conditions and find the maximum delay threshold. The shadow register has no functionality in the tested design other than reading information, in which case the delay measurement can be run online. [20]

Keeping in mind that the delays of positive and negative transitions in the PUM are not equal, the phase is shifted a total of 360° in as small steps as possible to gather the error data. The error detection trigger needs to be configured differently depending on which part of the sweep is under measurement due to a small blind spot in the measurement process. The trigger can be either the rising or the falling edge of the system clock or rising or falling edge of the test clock. [20]

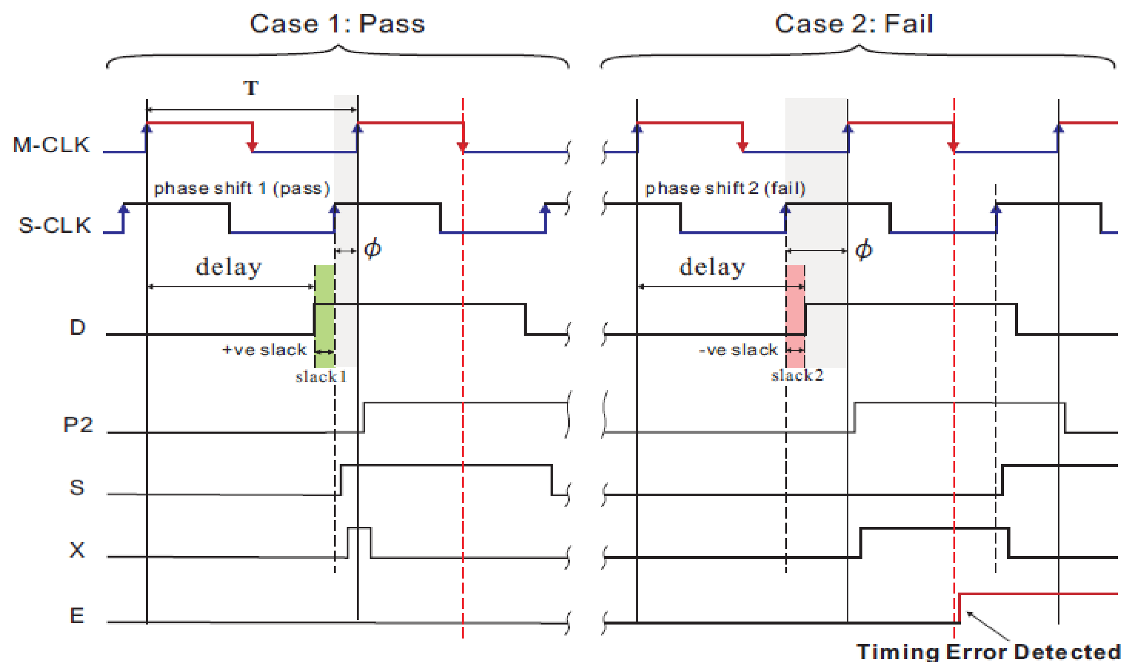


Figure 7: Timing diagrams of passing and failing online measurements [20]

The signals S and P2 need to be equal on the specified trigger or an error is reported. As seen on Figure 7, when the phase of the test clock goes past a threshold point, the shadow register starts to fail to transition its value according to D, causing S and P2 not to stay equal. The test may need to be run a maximum of four times with the different error trigger configurations to mitigate the effect of the blind spot, depending on the size of the blind spot. After a full measurement, a chart can be made with the failure rate against the amount of phase-shift. The slack values can be easily read from the chart.

The accuracy of this method is very dependent on the capabilities of the FPGA. The step-size resolution has a major influence on the accuracy but recent FPGAs should be able to get to a 100 ps resolution with a 100 MHz clock, which still gives decent accuracy for the delays on the nanosecond scale. [20]

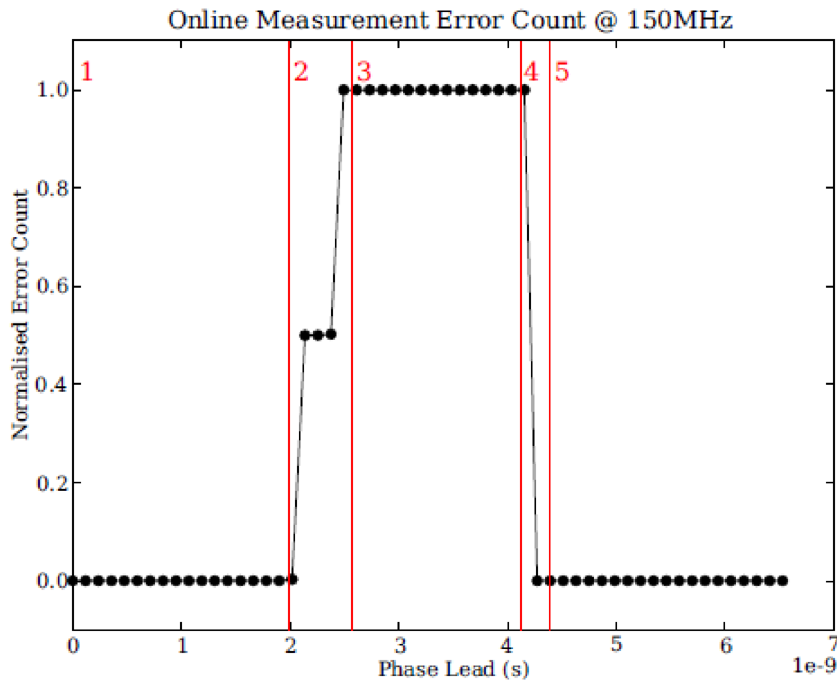


Figure 8: Charted results after 360° phase sweep on the test clock [20]

The chart of the results can be divided into different regions to analyze the operation of the design, as it is done in Figure 8. The third region has a failure rate of 100% due to the shadow register being a full clock cycle behind the system clock and making S and P2 signals go out of sync. The fact that the positive and negative transitions have different delays can be seen in the 50% threshold in the second region. The first region unveils the actual slack value, and by knowing the system clock frequency, the maximum delay of the path before malfunction can be calculated. [20]

To consider the accuracy, the blind spot on the fourth region might vary in size and will mitigate the measurement if it overlaps with the second region. Also, the condition of the path from the PUM to the shadow register will play an important role on the accuracy of the monitoring. If the path has suffered from

variations or starts to in the future, the results of the measurement cannot be trusted. This effect can be minimized by using the previous offline frequency-sweep method to evaluate the paths and by calibrating the online design accordingly. [20]

While temperature will slightly affect the measurement, the likely usage scenarios would suggest the device is past power-up phase and not quickly fluctuating in temperature. This gives the user some control over the measurement conditions to increase the accuracy.

Implementation of the design is feasible if the FPGA can fit the clock generation requirements. The design is very simple and requires a small area, while giving information about the condition of the wanted critical path on a design. A downside for the design is that it favors a high system clock frequency on the measured path to provide accurate results.

An idea to increase the accuracy especially in lower system clock speeds would be to use multiple phase-locked loops (PLL) in the test clock generation to possibly decrease the step size of the test clock generator. The PLLs could be set to slightly different phases and they could be activated in an order according to the state of the sweep to manage a phase-sweep with better step resolution.

A strongly recommended extra step is to evaluate the paths beforehand and calibrate the design to increase the accuracy. By configuring the shadow register to lead or lag the system clock, the effects of process variations on the PUM and path of the shadow register can be mitigated.

The design can easily be converted to the offline frequency-sweep design and back to the online phase-sweep design to give the most functionality to evaluate the wear-levels and process variations in FPGAs. In the offline measurement the accuracy is much greater and by implementing the parallel design it can be used to map the whole chip by its delay values. The online design enables decently accurate slack time measurement simultaneously with running another design on the FPGA. [20]

8 CONCLUSIONS

Scaling of the manufacturing processes will continue and recent developments have shown that the reliability and longevity in ICs are sacrificed in order to keep up with Moore's Law. FPGAs are more widely used and they too are falling into victims of the process variations and decreases in reliability in favor of the decreasing feature size. However, modern FPGAs come with a range of powerful features and with advanced clock generation resources, designs can be implemented on the FPGA to study the effects of process variation and degradation.

The traditional approach of measuring ICs using a ring oscillator-based design is not feasible to implement on to a FPGA due to the area cost and low accuracy. Instead, an offline frequency-sweep-based design using the clocking resources of the FPGA can be implemented to get results from singular or small groups of logical elements with feasible effort. By applying more effort, the design can be implemented in parallel to test large areas of the chip simultaneously. The area and resource requirements are low in this design.

Online measurement capability can be added to the design by modifying it slightly. The frequency-sweep is changed to a phase-sweep to give the ability to run the testing frequency at the same speed with the system clock, which enables the online measurement. Some accuracy is lost in the new method but it is still decent for the analysis. Feasible effort is needed for the implementation, but the accurate phase-sweep creates some requirements for the FPGA hardware.

Modern FPGAs carry the resources to implement a built-in self-test to accurately evaluate the reliability of the logic elements. The advances in future will most likely increase the accuracy of the measurements. Researching this field helps to better understand the development and operation of FPGAs and ICs in general.

REFERENCES

[1] How Products Are Made: Integrated Circuit. [Online Document] http://www.madehow.com/Volume-2/Integrated-Circuit.html (Accessed 11.5.2015)

[2] Intel. 2012. 3rd Generation Intel Core Processor Family Quad Core Launch Product Information [Online Document] http://download.intel.com/newsroom/kits/core/3rdgen/pdfs/3rd_Generation_Intel_Core_Product_Information.pdf (Accessed 11.5.2015)

[3] Wikipedia. Semiconductor device fabrication. [Online Document] https://en.wikipedia.org/wiki/Semiconductor_device_fabrication (Accessed 11.5.2015)

[4] Clayton M. Christensen, Steven King, Matt Verlinden, Woodward Yang. 2008. The New Economics Of Semiconductor Manufacturing. [Online Document] http://spectrum.ieee.org/semiconductors/design/the-new-economics-of-semiconductor-manufacturing (Accessed 12.5.2015)

[5] Center for Responsible Nanotechnology (CRN). What is Nanotechnology? [Online document] http://www.crnano.org/whatis.htm (Accessed 12.5.2015)

[6] Transistors go 3D as Intel re-invents the microchip. [Online Document] http://arstechnica.com/business/2011/05/intel-re-invents-the-microchip/ (Accessed 4.6.2015)

[7] Micross Components. 2008. Shrinking Silicon Feature Sizes [Online Document] http://www.micross.com/pdf/Micross_Technical_Presentation-Effect_of_Die_Shrinkage.pdf (Accessed 7.5.2015)

[8] Semelab. A Short Guide to Quality and Reliability Issues In Semiconductors For High Rel. Applications [Online Document] http://www.semelab-tt.com/uploads/q-and-r-in-hi-rel.pdf (Accessed 13.5.2015)

[9] Srinivasan, J.; Adve, S.V.; Bose, P.; Rivers, J.A., "The impact of technology scaling on lifetime reliability," Dependable Systems and Networks, 2004 International Conference on , vol., no., pp.177,186, 28 June-1 July 2004

[10] Srinivasan, J.; Adve, S.V.; Bose, P.; Rivers, J.A., "Lifetime reliability: toward an architectural solution," Micro, IEEE , vol.25, no.3, pp.70,80, May-June 2005

[11] Computer Simulation Laboratory (CSL). Electromigration. [Online Document] http://www.csl.mete.metu.edu.tr/Electromigration/emig.htm (Accessed 15.5.2015)

[12] University of Cambridge . Electromigration [Online Document] http://www.msm.cam.ac.uk/mkg/e_mig.html (Accessed 4.6.2015)

[13] J.R. Lloyd. Electromigration for Designers: An Introduction for the Non-Specialist. [Online Document] http://www.eetimes.com/document.asp?doc_id=1275855 (Accessed 15.5.2015)

[14] University of Cambridge. Dielectric breakdown [Online Document] http://www.doitpoms.ac.uk/tlplib/dielectrics/breakdown.php (Accessed 17.5.2015)

[15] Atomic World. Electron Tunneling. [Online Document] http://www.hk-phy.org/atomic_world/stm/stm02_e.html (Accessed 17.5.2015)

[16] Wikipedia. Process Variation (Semiconductor). [Online Document] https://en.wikipedia.org/wiki/Process_variation_%28semiconductor%29 (Accessed 19.5.2015)

[17] Xilinx. What is a FPGA. [Online Document] http://www.xilinx.com/fpga/index.htm (Accessed 28.4.2015)

[18] Wong, J.S.J.; Sedcole, P.; Cheung, P.Y.K., "Self-characterization of Combinatorial Circuit Delays in FPGAs," Field-Programmable Technology, 2007. ICFPT 2007. International Conference on , vol., no., pp.17,23, 12-14 Dec. 2007

[19] National Instruments. Introduction to FPGA Technology: Top 5 Benefits. [Online Document] http://www.ni.com/white-paper/6984/en/ (Accessed 10.6.2015)

[20] Levine, J.M.; Stott, E.; Constantinides, G.A.; Cheung, P.Y.K., "Online Measurement of Timing in Circuits: For Health Monitoring and Dynamic Voltage & Frequency Scaling," Field-Programmable Custom Computing Machines (FCCM), 2012 IEEE 20th Annual International Symposium on , vol., no., pp.109,116, April 29 2012-May 1 2012

[21] Anindo Mukherjee, Kevin Skadron. Measuring Parameter Variation on an FPGA Using Ring Oscillators. [Online Document] http://www.cs.virginia.edu/~skadron/Papers/pv_tr2006_16.pdf (Accessed 18.5.2015)