Inducing Non-uniform FPGA Aging Using Configuration-based Short Circuits

HAYDEN COOK, JACOB ARSCOTT, BRENT GEORGE, TANNER GASKIN, JEFFREY GOEDERS, and BRAD HUTCHINGS, Brigham Young University, USA

This work demonstrates a novel method of accelerating FPGA aging by configuring FPGAs to implement thousands of short circuits, resulting in high on-chip currents and temperatures. Patterns of ring oscillators are placed across the chip and are used to characterize the operating frequency of the FPGA fabric.

Over the course of several months of running the short circuits on two-thirds of the reconfigurable fabric, with daily characterization of the FPGA 6 performance, we demonstrate a decrease in FPGA frequency of 8.5%. We demonstrate that this aging is induced in a non-uniform manner. The maximum slowdown outside of the shorted regions is 2.1%, or about a fourth of the maximum slowdown that is experienced inside the shorted region. In addition, we demonstrate that the slowdown is linear after the first two weeks of the experiment and is unaffected by a recovery period.

Additional experiments involving short circuits are also performed to demonstrate the results of our initial experiments are repeatable. These experiments also use a more fine-grained characterization method that provides further insight into the non-uniformed nature of the aging caused by short circuits.

CCS Concepts: • **Hardware → Reconfigurable logic and FPGAs**; • **Security and privacy → Security in hardware;**

Additional Key Words and Phrases: Aging, short circuits, NBTI, PBTI, electromigration, hardware security, FPGA

ACM Reference format:
Hayden Cook, Jacob Arscott, Brent George, Tanner Gaskin, Jeffrey Goeders, and Brad Hutchings. 2022. Inducing Non-uniform FPGA Aging Using Configuration-based Short Circuits. *ACM Trans. Reconfigurable Technol. Syst.* 15, 4, Article 41 (June 2022), 33 pages.
https://doi.org/10.1145/3517042

1 INTRODUCTION

Through normal operation of semiconductor devices, the performance characteristics of transistors gradually decline, resulting in decreased maximum clock speeds. This performance degradation, referred to as *transistor aging*, is the result of several physical mechanisms (***negative bias temperature instability (NBTI), electromigration (EM)***, and more), and is generally a greater concern when scaling to smaller technology nodes [1]. **Field-programmable gate arrays (FPGAs)** are not immune to this effect, and several studies have measured how FPGA performance is affected by transistor aging [2–4].

Understanding FPGA aging is critical. Of highest importance, FPGA design tools must account for aging mechanisms to ensure the reported f_{max} is achievable over the entire lifetime of a part.

Authors' address: H. Cook, J. Arscott, B. George, T. Gaskin, J. Goeders, and B. Hutchings, Brigham Young University, EB 450, Provo, UT, 84602; emails: {hcook95, jarscott, brent_george, gaskin.tanner, jgoeders, brad_hutchings}@byu.edu.
Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2022 Association for Computing Machinery.
1936-7406/2022/06-ART41 $15.00
https://doi.org/10.1145/3517042

Aside from this, however, aging does play a role in other aspects of FPGA tools. In Reference [4], Dogan et al. show that FPGA aging models can be used as an effective predictor of recycled parts, and in Reference [5], Maiti et al. demonstrate how aging disrupts the reliability of **physical unclonable functions (PUFs)**. Several works have shown how aging can be rapidly induced, inflicting years of wear-out on parts in a short time period [2, 5]. This is typically accomplished by raising the supply voltage above normal levels and baking the part in high temperatures.

In this work, we induce accelerated FPGA aging by loading several FPGAs with different bitstreams containing thousands of short circuits. Each FPGA is configured with a different shorted design, and once loaded, can cause the FPGA devices to sink currents up to 7.9 A and self-heat to temperatures over 170 °C. During a typical experiment, the FPGA is left in this stressed state for a period of 24 hours. Afterwards, characterization bitstreams are loaded onto the FPGA to determine how much additional delay has been induced by the excessive heat and current. The characterization bitstreams use **ring oscillators (ROs)** to measure relative increases in delay induced by the accelerated aging process. Aging experiments repetitively load the shorting and characterization bitstreams to measure the accelerated aging process over periods of days or weeks.

Our approach is unique in that it does not require modifications to the supply voltage or an external heat source. This technique could be used to potentially perform aging remotely without physical control of the FPGA. This approach, however, would require that the system be equipped with a relatively high-amperage power supply.

Of perhaps more interest, our technique can apply aging non-uniformly across the die, something that is not possible with previously published techniques. To our knowledge this is the first work that has demonstrated targeted, nonuniform aging on any commodity semiconductor device, not just FPGAs. In this work, we present several experiments that explore different patterns of placing short circuits on FPGA chips, exploring both the overall impact of using short circuits to perform device aging, as well as exploring the granularity to which we can cause aging in specific regions of the FPGA using these targeted methods.

We believe that as FPGAs continue to be used in more applications, including safety-critical IoT devices and environments where users can deploy FPGA bitstreams on remote machines (Amazon EC3), understanding and planning for these aging techniques will be of high importance.

The main contributions and novelty of this work are:

- Demonstrating how FPGA bitstreams containing short circuits can induce accelerated FPGA aging, including one long-running experiment that achieved a slowdown of up to 8.5%.
- Several experiments that demonstrate non-uniform aging across FPGA parts, including a variety of short circuit patterns to explore the limits of trying to perform aging on a small targeted region of the FPGA.
- A framework for performing controlled, fine-grained characterization of the FPGA fabric speed.
- Demonstrating that our techniques do not experience any recovery; the induced aging appears to be permanent.

In this article, Section 2 discusses background material, Section 3 presents our techniques to measure aging, Section 4 presents our techniques to induce aging, and Section 5 presents our experimental methodology and results. Section 6 discusses various observations across different experimental designs, and Section 7 discusses conclusions and future work.

2 BACKGROUND

This section discusses aging mechanisms that are relevant for FPGAs, as well as a discussion of related work on inducing FPGA aging.

2.1 Aging Mechanisms

Prior work has established several aging mechanisms pertinent to CMOS technologies: **bias temperature instability (BTI), electromigration (EM), hot carrier injection (HCI)**, and **time-dependent dielectric breakdown (TDDB)** [6–10].

2.1.1 Bias Temperature Instability (BTI). BTI is a common problem in MOSFET technology and has become prevalent as technology nodes have shrunk [1]. There are two distinct cases of BTI: **negative-bias temperature instability (NBTI)**, which affects PMOS transistors in CMOS, and positive bias temperature instability, which affects NMOS transistors in CMOS. In older CMOS technology nodes, only NBTI was a concern. With the introduction of high-k dielectrics in sub-45 nm technologies, however, PBTI has started to become as much of a concern as NBTI [11, 12].

BTI occurs when an electric field is applied across the gate oxide of MOSFET transistors. This can create interface traps and fill in preexisting traps with carriers in the dielectric. This phenomena causes the threshold voltage to slowly increase over time [6, 9]. As the threshold voltage increases, the switching speed of the transistors decrease. This effect is more pronounced with higher temperatures and voltages [10, 13].

2.1.2 Electromigration (EM). EM is the gradual migration of metal atoms in wires over time due to high electric current density. As this migration occurs, the metal atoms will slowly accumulate at one end of the channel, thus narrowing the wire [6]. This may result in slower switching speeds along the affected wires due to increased resistances, but is usually associated with chip failure due to the formation of an open circuit along the wire or a short circuit with another wire [6, 14]. EM is accelerated by both high DC current and high temperatures [10, 14], both of which are the primary stressors introduced by our aging technique.

2.1.3 Hot Carrier Injection (HCI). HCI occurs when carriers gain sufficient kinetic energy to overcome a potential barrier and are injected into the gate oxide layer of a transistor. As hot carriers are injected into the oxide, the chemical bonds of the oxide break down, creating interface traps, which raises the threshold voltage and lowers switching speeds [6, 7]. This aging effect intensifies as voltage and high substrate current increases [9, 10, 15]. While the temperature dependence of HCI is not yet conclusive, it is often seen that the effects of HCI slightly decrease with increased temperature [16, 17].

2.1.4 Time-Dependent Dielectric Breakdown (TDDB). TDDB occurs when defects created in the gate oxide cause traps to accumulate, creating a conductive path between the substrate and gate [7–10]. This increases the leakage current of the transistors, which raises the overall power consumption and lowers switching speeds [13]. Eventually, TDDB can cause a hard breakdown of the dielectric and lead to device failure [6]. Defects that contribute to TDDB occur under constant stress, thus increasing device voltage and accelerating the effects of TDDB. However, degradation caused by TDDB decreases with increased temperature [18].

2.2 Related Work

Our work is not the first to introduce short circuits in the FPGA bitstream, nor is it the first to induce FPGA aging; however, it is the first to combine the two ideas, resulting in the ability to perform localized aging via configuration.

Beckhoff et al. demonstrate how early partial reconfiguration tools can result in short circuits that increase current and power consumption [19]. Hadzid et al. discuss how short circuits in FPGAs can cause them to operate in an unsafe operating region, potentially exceeding the power supply capability and disrupting operation, or perhaps even damaging the part, though actual

damage was not shown [20]. Hutchings et al. show that short circuits can intentionally be inserted into FPGAs and that the relationship between the the number of short circuits and power consumption is highly linear [21].

Previous aging studies have primarily used a combination of over-voltage and high temperatures to induce NBTI and HCI to accelerate aging [5, 10, 22]. Stott et al. raise the voltage from 1.2 V to 2.2 V and increase the temperature to 147 °C to accelerate aging on an Altera Cyclone III device. They observe a 15% slowdown over 75 days with this method [10]. Maiti et al. demonstrate a slowdown on a Xilinx Spartan 3e device using two stress phases. The first increases voltage from 1.2 V to 1.5 V and the temperature to 70 °C, producing a slowdown of approximately 5.0% after 200 hours. The second increases the voltage to 1.8 V and the temperature to 80 °C, which resulted in a total slowdown of approximately 6.7% after an additional 200 hours. Between the two phases, they have a one-day recovery period where the slowdown recovers half a percentage point, from 5.0% to 4.5% [5]. Slimani et al. demonstrate aging on a Xilinx Artix-7 part. This is the same FPGA family that we use in our experiments. They achieve a 1.8% slowdown by increasing the temperature to 125 °C for 14 days. They have an eight-day recovery period where the slowdown recovers 1 percentage point, from 1.8% to 0.8%. [22].

As far as we are aware, only one other previous work has discussed a configuration technique that solely uses configuration to produce a slowdown. Chakraborty et al. discuss the idea of using ring oscillators to heat up the chip and cause a slowdown, but do not validate their model through experimentation [23].

Additionally, to measure aging, the intrinsic speed of the FPGA fabric must be characterized. Many techniques to accomplish this have been proposed. Zick et al. propose using low-cost **ring oscillators (ROs)** that output to a residue number system [24]. Hung et al. propose a time-to-digital converter that uses shadow registers to precisely measure individual wires and logic [25]. As discussed in the next section, we have decided to use counter-based ring oscillators to characterize the FPGA.

3 CHARACTERIZING FPGA FABRIC SPEED

This section describes the techniques used to characterize the performance of the FPGA fabric, both before and after artificially aging the FPGA.

3.1 Ring Oscillators

We use ROs to test the intrinsic speed of the FPGA fabric and measure the effect of short circuits in causing device degradation. ROs are purely combinational circuits that are built using a ring of an odd number of inverters, causing a pulse to travel around the ring at the maximum speed supported by the FPGA fabric.

Figure 1 shows the design of the ROs used in our experiments. As shown, the ROs contain a single AND-gate that is used to disable and enable the oscillation using the *ro_en* signal. A wire is tapped off of the RO, labelled *ro_clk*, that is then fed into the clock input of a large counter. By running the RO for a set amount of time, and observing the counter value, we can calculate the RO oscillation frequency.

3.2 Creating Characterization Bitstreams with RapidWright and Vivado

We use RapidWright [26] to create custom placed ring oscillators in the FPGA fabric. RapidWright is an open-source tool from Xilinx that can manipulate netlists to provide very fine placing and routing control. It interfaces with Vivado via design checkpoints.

Our characterization bitstreams are responsible for measuring the intrinsic delay of the FPGA fabric before and after short circuit burns. They include a static region, which contains a

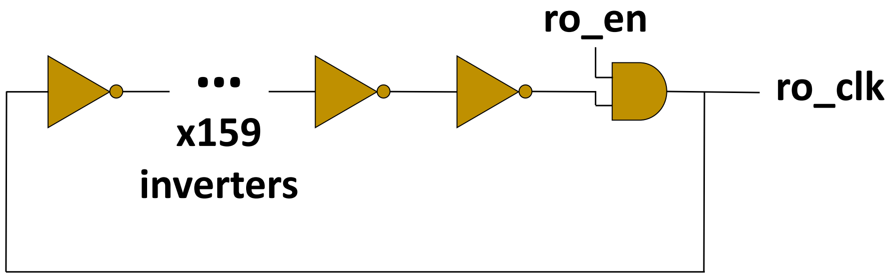


Fig. 1. Diagram of a *Ring Oscillator* with a *ro_en* enable input signal and a *ro_clk* output signal that feeds into a counter.

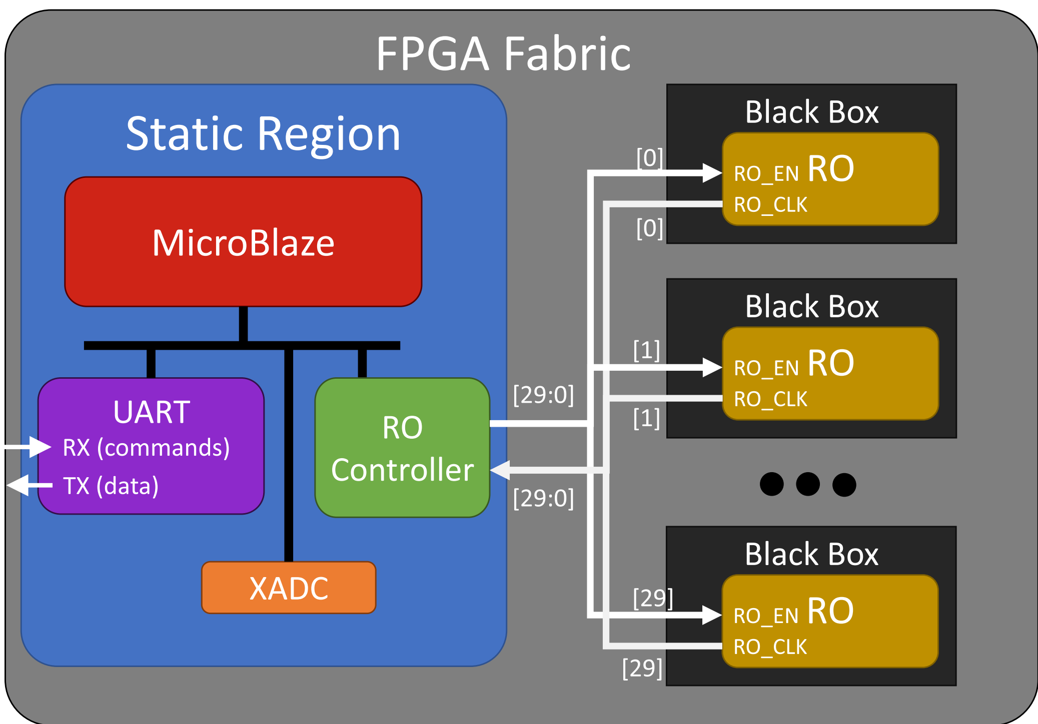


Fig. 2. Diagram showing the general layout of our characterization bitstreams. The position of the static region and ROs move around in different bitstreams, allowing us to characterize the whole chip. While the number of ROs varies from bitstream to bitstream, we are limited by the number of clock lines to at most 30 ROs for a single bitstream.

MicroBlaze, custom control hardware with hardware timers, and UART hardware. They also contain black boxes that allow for the insertion of the ROs.

A diagram of the general layout of our characterization bitstream can be seen in Figure 2. Each black box contains a RO partial design containing a single RO as described earlier. The *ro_clk* signals are connected to counters in the custom RO controller via clock lines. Since the FPGA contains only 32 clock lines, two of which are used by our static design, we are limited to 30 ROs per bitstream. However, it is unlikely that using a clock line for the *ro_clk* signal has any impact on

the RO frequency. Future experiments may explore using other routing resources, allowing more ROs to fit within a single characterization bitstream.

The RO controller also has control of the the *ro_en* signals. This allows the custom hardware to enable the RO and counter for an exact number of cycles, measured by a timer within the RO controller. This ensures that characterization is always performed for the exact same duration of time and provides an accuracy that far exceeds the variability of the RO's frequency.

The RO controller is overseen by an on-chip MicroBlaze processor running custom software. The MicroBlaze can receive commands over UART, giving us external control of the experiment during runtime. The MicroBlaze also collects and processes RO data from the RO controller and environmental board data from the on-board XADC. It then sends this data over the UART so it can be recorded into a database.

While the static region is designed in Vivado, the bitstreams are created using a combination of Vivado and RapidWright tools. For each RO, a partial design is created in RapidWright, which allows us to create a design that contains specific placement of RO elements. RapidWright also gives us control over the logical netlist that represents how the placed RO elements should be connected together. The unrouted RO designs are then imported into Vivado by inserting them into black boxes. The static region is then synthesized and implemented along with the pre-placed ROs, and a bitstream is generated.

While it would be possible to manually pre-route the ROs in RapidWright instead of using Vivado's router, we decided it would not be practical. Our characterization bitstreams contain hundreds of ROs and, since the FPGA is not regular enough to use a single routing pattern that can be applied across the entire chip, each of the ROs would have to be individually hand-routed, which is impractical.

While another option would be to write our own router in RapidWright for our ROs, it is simpler to just have Vivado perform the RO routing. Although we were worried that using Vivado's router would result in some irregularity between ROs, upon inspection, it appeared that all ROs had very similar routing patterns. This visual inspection was reinforced by a RapidWright script that analyzed the RO routing and determined that in all cases the routing from one LUT to the next uses the same number of PIPs, and routing always remains within the source and neighboring destination tiles (including the adjacent switchbox tiles). Furthermore, even if there is some variation between ROs, our experiments are measuring slowdown, i.e., *change* in frequency, so small variations in initial frequencies will not impact our results.

3.3 Characterization Methods

3.3.1 Characterizing with Three ROs. For our *High Density Bottom Experiment* (originally published in Reference [27] and discussed later in Section 4.3.1), the technique for characterizing the FPGA fabric throughout the experiment is to measure the frequency of distinct ROs one at a time. This technique uses three ROs, as seen in Figure 3. These ROs are 160 nodes long (159 inverters and an AND gate) and are placed at the top, bottom, and center of the chip to measure the speed of the FPGA fabric before and after introducing the short circuits. Each RO is routed with the serpentine pattern shown in Figure 4. This routing pattern is intended to equalize the delay between each node as much as possible and to use a mix of local and global routing.

Each RO has its own bitstream and they are configured and measured one at a time. We spend 20 minutes measuring the frequency of each RO, during which the RO frequency is sampled once per second. The first seven minutes of data is ignored to allow temperature fluctuations to settle after a burn period.

3.3.2 Characterizing with a RO Heatmap. While the *Three RO* characterization method is sufficient to detect aging at different locations of the chip, we quickly realized that we wanted much

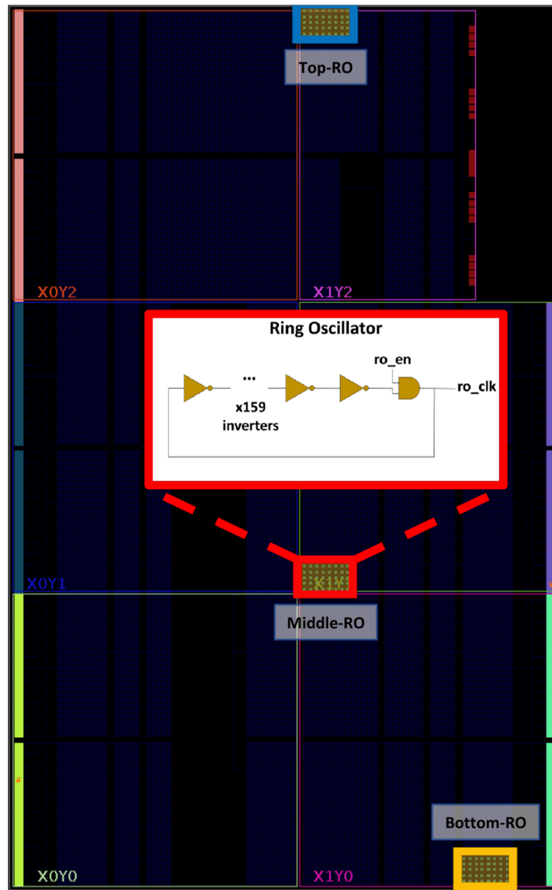


Fig. 3. The layout of the FPGA showing the location of the three distinct ROs used to characterize the *High Density Bottom Experiment*. Although this figure shows the ROs on the same layout, the ROs are programmed and characterized separately.

more fine-grained resolution into the performance of the FPGA fabric throughout an experiment. To do this, we developed an improved technique to substantially increase the number of ROs we use during characterization. By spreading these ROs across the entire chip, we create a heatmap of FPGA fabric speeds before and after any aging process. Such a heatmap showing the initial raw frequencies of one of our FPGAs (used in our *Initial Checkerboard Experiment*) can be seen in Figure 5.

Each RO in the heatmap measures an area that is four slices wide and five slices tall, again using the serpentine pattern shown in Figure 4. This gives the ROs a size of 80 nodes, which is half as large as the three ROs used in the previous method. The size was chosen to prevent any RO from crossing clock boundaries, BRAM columns, or DSP columns while still limiting the number of bitstreams needed to characterize the entire chip.

Since 7-series chips only have 32 clock lines, two of which are used by our static region, each heatmap bitstream can only contain up to 30 ROs. The ROs in each bitstream form a single column that spans the height of the FPGA. While we were initially worried that measuring several running ROs in close proximity to each other would create self-heating effects and degrade the accuracy of the results, further testing determined this was not an issue.

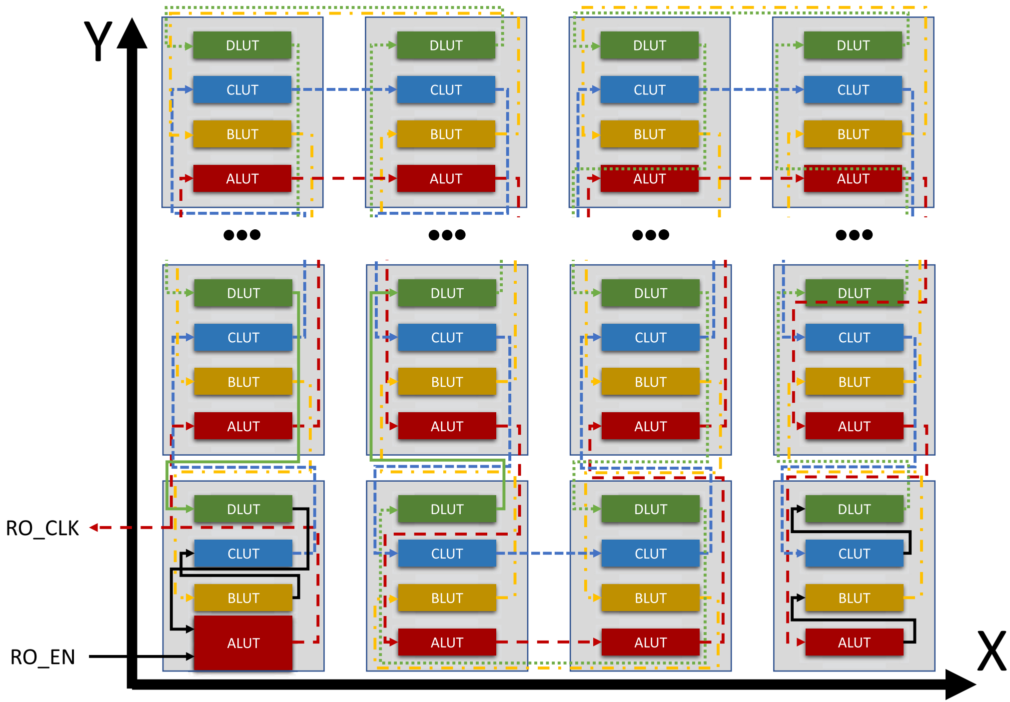


Fig. 4. Diagram showing the serpentine routing of our ROs. All ALUTs in an area are chained together and then connected to the chain of BLUTs, which in turn are connected to the chain of CLUTs, and finally the chain of DLUTs. The three ROs of our initial characterization method use a 160-node RO, covering a range of slices five tall and eight wide. The heatmap characterization method uses 420 ROs, where each RO is 80 nodes, covering a range of slices five tall and four wide.

To get a heatmap of the entire FPGA, we need to create several heatmap bitstreams to cover all regions of the FPGA. In total, our heatmap characterization consists of 420 different ROs, spread across 17 heatmap bitstreams (not every bitstream contains 30 ROs in the column, since some FPGA logic columns are partial columns).

During characterization, the frequency of all ROs in a bitstream are simultaneously measured for 10 seconds, taking approximately eight measurements per second. We decreased the length of measurement from 13 minutes to only 10 seconds after determining that measuring for longer did not improve the standard deviation of our measurements.

In addition to shortening the measurement time of the heatmap ROs, we also decided to increase the amount of cooldown time that takes place prior to the characterization process. While the *Three RO* approach waits only 7 minutes before collecting characterization data, we decided to increase this wait time to 40 minutes for our *heatmap approach*. This choice was made after analyzing the data from the *Three RO* characterizations and finding that a small amount of cooling still takes place after the 7-minute cooldown period. While the benefits to an increased cooldown time are fairly minor, we only characterize once a day, so the additional wait time was acceptable.

4 FPGA AGING USING SHORT CIRCUITS

4.1 Implementing Short Circuits

We accelerate the aging of FPGAs by introducing short circuits into the FPGA fabric. A short circuit, in the context of FPGA designs, is when the routing is configured such that multiple drivers

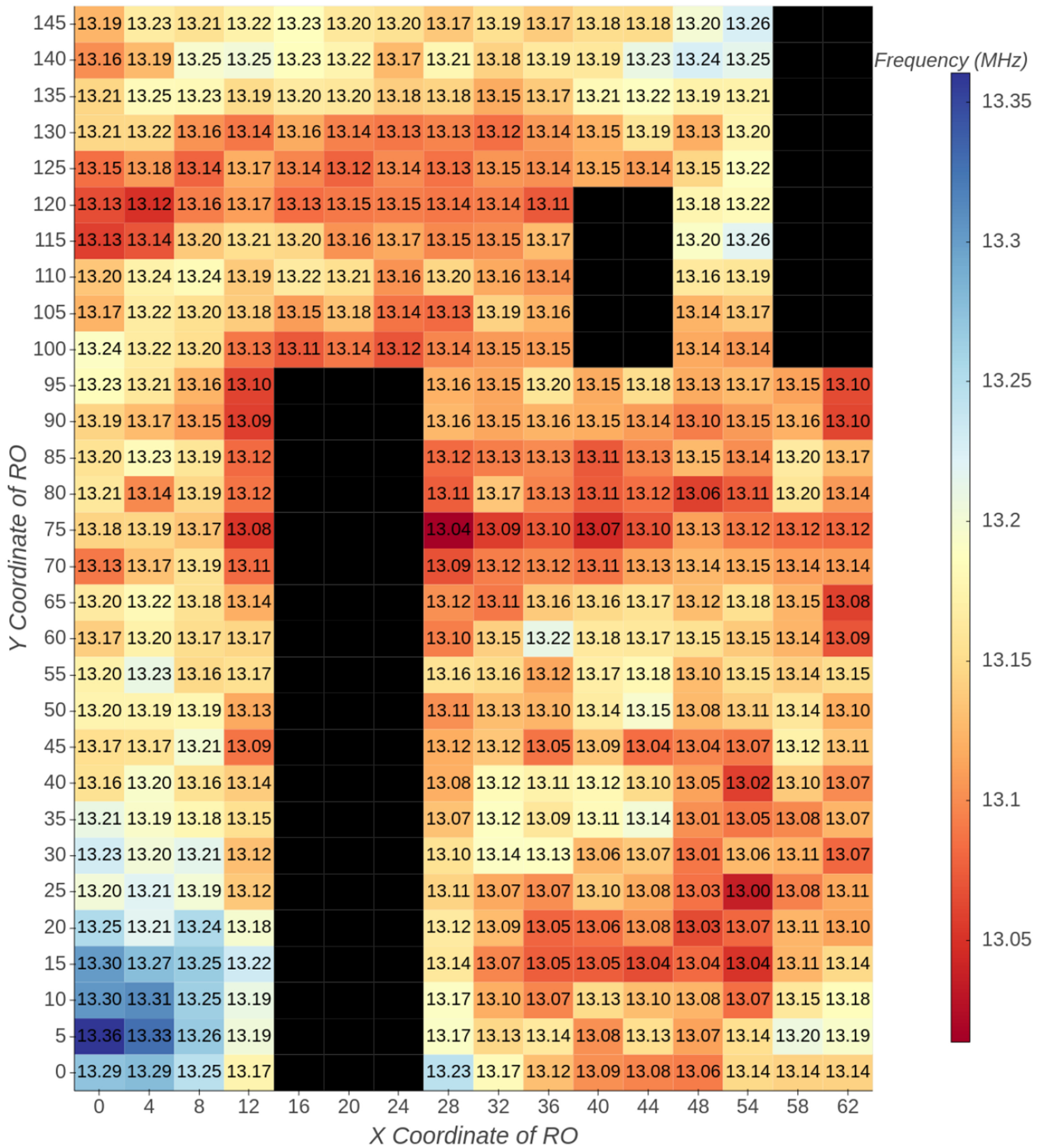


Fig. 5. A heatmap showing the initial frequencies of the *Initial Checkerboard Experiment*.

(one logic-1 and one logic-0) drive the same net. Our approach, which can be seen in Figure 6, implements a short circuit by using the **lookup table (LUT)** and **flip-flop (FF)** primitives, since they can easily be configured to drive a constant logic-0 or logic-1. We then select two of these primitives that connect to the same routing multiplexer, and the two multiplexer inputs are both activated (in legal circuits only one input to a routing mux would ever be active at a time).

Implementing thousands of short circuits simultaneously in the FPGA fabric causes high current flow, which in turn causes temperatures to rise well past safe thresholds. Due to the drastic increase in temperature, we use the term "burn" to indicate programming short circuits onto the FPGA to accelerate aging.

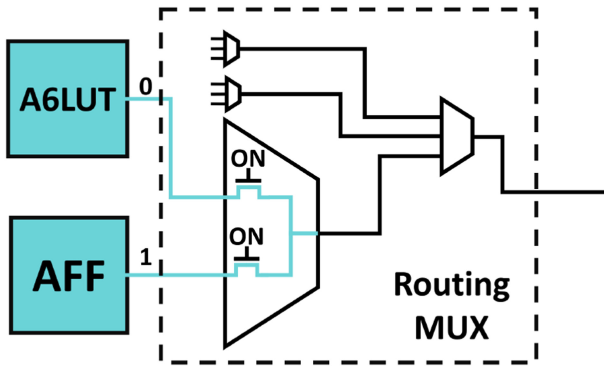


Fig. 6. A diagram of a short circuit implemented on an FPGA.

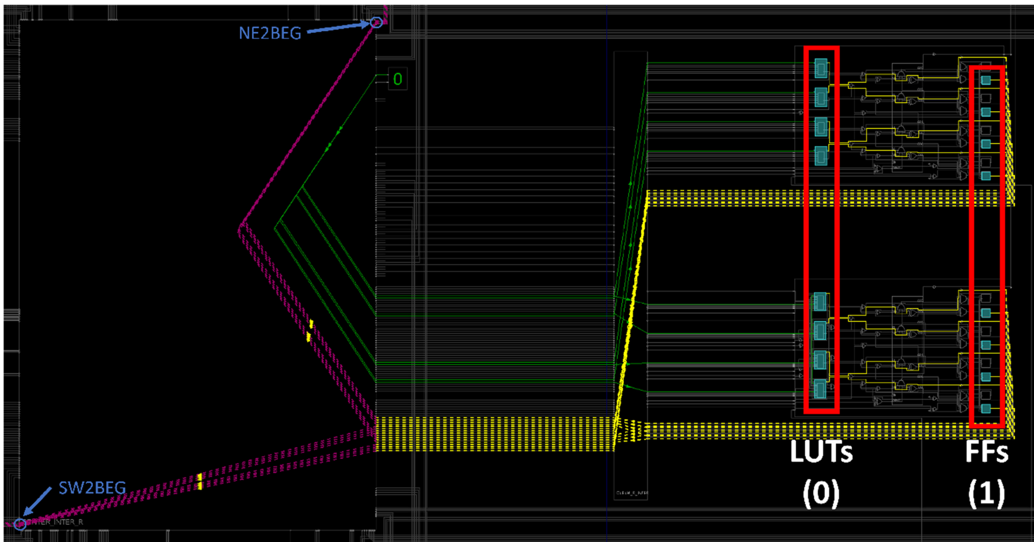


Fig. 7. A tile from one of our shorted designs. This tile contains eight LUT-FF pairs. The output of each LUT is set to be a constant logic-0, while the initial value of each FF is set to logic-1 and never changed. As with all of our experiments, the short circuits in this tile use the NE2BEG and SW2BEG PIP junction groups to complete the short.

Figure 7 shows the layout of a shorted CLBLM_R tile, which contains eight short circuits (four per slice). The LUTs and FFs are shorted together using the NE2BEG and SW2BEG PIP junction groups. These two groups are used in all of our experiments to create short circuits. We chose these PIP junctions early in our work, as our testing determined these routing paths provided relatively high current draw.

More recently, however, we conducted an exhaustive search to determine which PIP junction groups drew the most current. To do this, we created a bitstream for every PIP junction that could be shorted by a LUT and FF in the same site. Each bitstream contained 1,125 shorts all using the same type of PIP junction. We then loaded each bitstream onto a chip and used the on-board XADC via sysmon to measure the current at steady state. As shown in Figure 8, the NN2BEG and SS2BEG PIP junction groups drew the most current, a 7.16% increase over the PIP junctions we

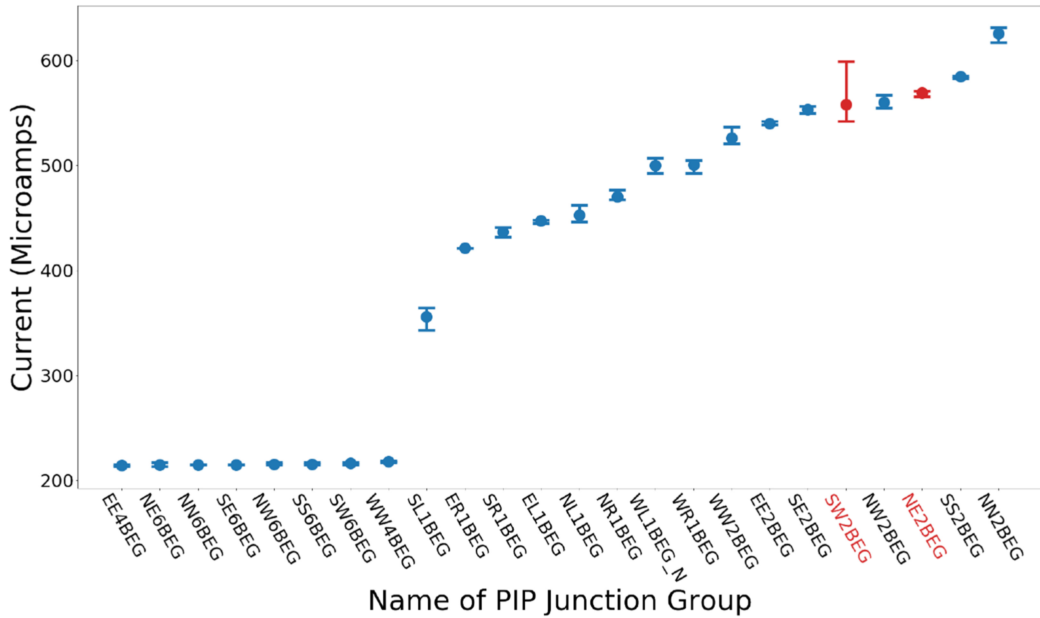


Fig. 8. This graph shows the average current of each shortable PIP junction group in increasing order. The currents were created by using bitstreams containing 1,125 shorts. Each bitstream used a single type of PIP Junction for each short. The two PIP groups we use to create short circuits in our experiments are colored red.

originally chose. However, to keep all of our experiments consistent, we decided to continue to use the non-optimal NE2BEG and SW2BEG PIP junction groups.

4.2 Creating Short Circuit Bitstreams

In this article, we present results for several different experiments where we implement thousands of these short circuits in different patterns on the FPGA. Each of these experiments uses a separate bitstream that implements these short circuits. These "burn" bitstreams exclusively contain short circuits and do not contain any of the characterization circuitry described in Section 3.

The design process of implementing these short circuits is completed using custom scripts in RapidWright. RapidWright allows us to individually enable the PIPs we need to create a short circuit, thus letting us bypass Vivado's restriction on routing a net with multiple drivers. Once the short circuit design is created in RapidWright and saved to a design checkpoint, it is then loaded into Vivado (version 2018.3). We then turn off all **design rule checks (DRCs)**, which allows Vivado to ignore our shorted nets and generate a bitstream that contains all of the short circuits.

It is entirely possible that a future version of Vivado may prohibit generating bitstreams with DRCs disabled. In this case, however, it is likely still possible to create these bitstreams with open-source CAD tools such as Symbiflow's FASM FPGA Assembler.

The Rapidwright short circuit tools we use to create the short circuit designs have been released on Github. They can be found at https://github.com/byuccl/short_circuit_aging.

4.3 Short Circuit Experiment Layouts

This article details our experimental testing with four different short circuit layouts; these layouts are described in this section.

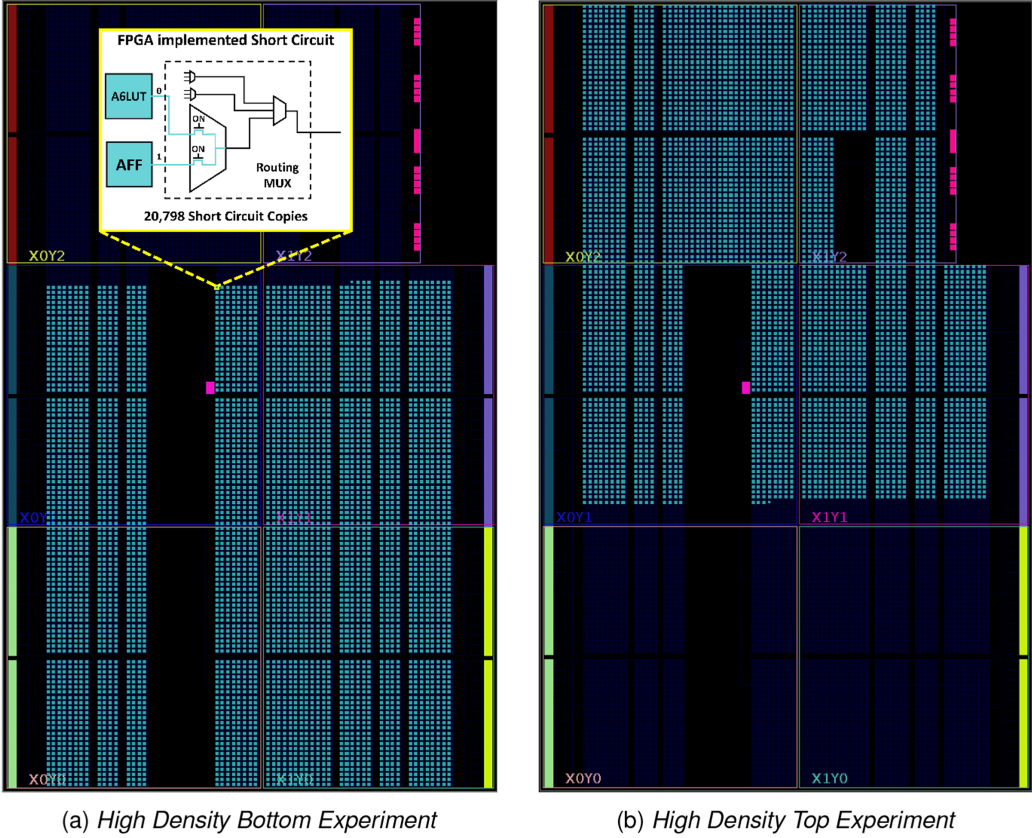
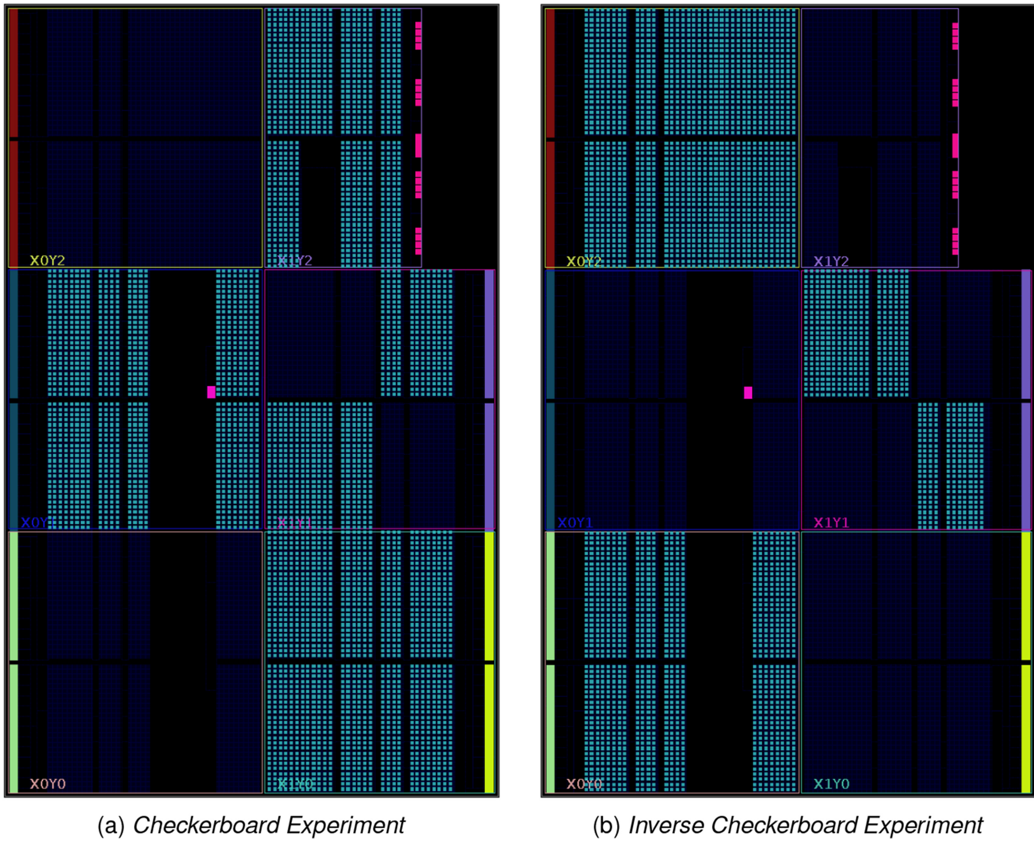


Fig. 9. Short circuit layouts for the two high-density experiments. Two-thirds (top or bottom) of the chip are shorted with 20,798 short circuits, while the other one-third remains blank.

4.3.1 High Density Bottom Experiment. The *High Density Bottom Experiment* (originally published in Reference [27]) uses a bitstream containing 20,798 short circuits, all of which are configured on the bottom two-thirds of the chip. The layout of this bitstream can be seen in Figure 9(a). To get the highest concentration of shorts, each slice in the shorted region contains four short circuits; this uses all of the LUTs within each slice and is the maximum density of short circuits we can create. Note that the dark black regions do not contain re-configurable logic and thus cannot implement short circuits.

It is worth noting that the part we used in our experiments (XC7A35TICSG324-1L) is a software-limited part. The physical layout is identical to the 50T part variants, but Vivado limits you to only using 20,800 of the 32,600 LUTs. Thus, while Figure 9(a) shows that the physical layout is empty in the top-third of the part, we implemented nearly the maximum number of short circuits possible for this part.

4.3.2 High Density Top Experiment. The layout for the *High Density Top Experiment* can be seen in Figure 9(b). In this experiment, 20,798 short circuits are placed on the top two-thirds of the chip. While very similar to the previous experiment, this experiment explores several additional aspects of short circuit aging. First, this experiment ensures that the difference in slowdown between the shorted and non-shorted regions is not due to the location of the shorted region, but is caused by



(a) *Checkerboard Experiment* (b) *Inverse Checkerboard Experiment*

Fig. 10. Short circuit layouts for the two checkerboard experiments.

the shorts themselves. Since we do not know the physical layout of the chip, we were unsure, for example, whether heat spreads the same way in the top part of the chip as the bottom. Second, the newly developed *Heatmap Characterization* method allows us to repeat the experiment with a more fine-grained understanding on how slowdown varies throughout the chip. Finally, since every chip produced naturally varies due to process variation, this experiment ensures that repeating a similar process on another chip yields similar results.

4.3.3 Checkerboard Experiment. The layout for the *Checkerboard Experiment* can be seen in Figure 10(a). This layout breaks up the large shorted region into much smaller shorted regions and places them diagonal from each other, similar to a checkerboard. Large regions of short circuits are placed in the *X0Y1*, *X1Y0*, and *X1Y2* clock regions. A smaller checkerboard pattern is also created within the *X1Y1* region. This layout contains 17,600 short circuits.

The initial results of the *High Density Bottom Experiment* demonstrate that there is a substantial (∼4×) difference in aging between the short circuit region in the bottom two-thirds of the chip and the blank region in the top-third [27]. The *Checkerboard Experiment* allows us to test if smaller, more complex regions of short circuits see similar localized aging effects. This experiment uses the *Heatmap Characterization* method, which allows us to observe how the slowdown caused by short circuits is distributed within the several shorted regions, as well as how quickly the slowdown drops off as we move away from the shorted regions.

4.3.4 Inverse Checkerboard Experiment. The *Inverse Checkerboard Experiment* is the direct inverse of the *Checkerboard Experiment* in that every shorted region in the former is not shorted in the latter and vice versa. This layout contains 14,900 shorts and can be seen in Figure 10(b).

Like the *High Density Top Experiment*, this experiment allows us to demonstrate repeatability on another part, testing a different physical location on the chip. It also allows us to investigate the effect the non-configurable regions (denoted as black regions on the layout) have on the spread of the damage caused by short circuits.

5 EXPERIMENTAL METHODOLOGY AND RESULTS

This section describes our experimental aging process—which consists of repeatedly burning and characterizing the chip—and presents results from our experiments.

5.1 Experimental Process

Each of our experiments includes at least one burn phase, which iteratively sequences through the following steps:

(1) Load the characterization bitstreams to obtain an initial *pre-burn* characterization of the chip.
(2) Load the short circuit configuration onto the FPGA and allow the FPGA to "burn" in this configuration for 24 hours.
(3) Load the characterization bitstream to determine increases in circuit delay.
(4) Repeat Steps 2 and 3 for several days (length varies by experiment).

In addition to a burn phase, three of our experiments include a recovery phase to test if the degradation caused by the burn recovers when stress is eliminated. The recovery phase for each experiment uses the same steps as the burn phase, but instead of loading a short circuit configuration onto the FPGA, a blank configuration is loaded for 24 hours.

5.2 Equipment Setup and Data Analytics

Measuring changes to circuit delay can be challenging, since the RO frequencies are not only sensitive to FPGA aging, but are also highly sensitive to both the supply voltage and ambient temperature. To guarantee that our measurements accurately reflect aging and are not influenced by voltage fluctuations or changes in ambient temperature, we created a test bed that allows us to have precise control over and detailed measurement of both supply voltage and ambient temperature.

Figure 11 illustrates our test setup. To control environmental temperature, all boards were placed within a thermal chamber during the entire experiment process. A temperature sensor, located inside the chamber, was monitored using an Intel NUC mini PC and LabJack ADC, and the measurements were recorded in a MySQL database multiple times per second. When set to 35 °C, this chamber kept the ambient temperature of the boards at 35 ± 0.086 °C. While the beginning and end of each burn period process saw a sudden, but small (0.1 °C), change in ambient temperature, the temperature would always stabilize to the nominal 35 °C within three minutes.

Note that the thermal chamber only provides a stable ambient temperature and does not play a role in the accelerated aging process.

The test boards were Arty A7-35T development boards with an Artix XC7A35 FPGA. These boards were not previously used to run any other designs prior to our aging experiments. A different Arty board was used for each experiment, meaning four different boards were used in this work. The number of experiments performed was mainly limited by the time it took to perform the experiments and our limited test equipment.

The boards were modified such that the $VINT$ supply voltage to each FPGA was supplied by either an E36231 or an N6705C Keysight power supply. Notably, the external power-supply could

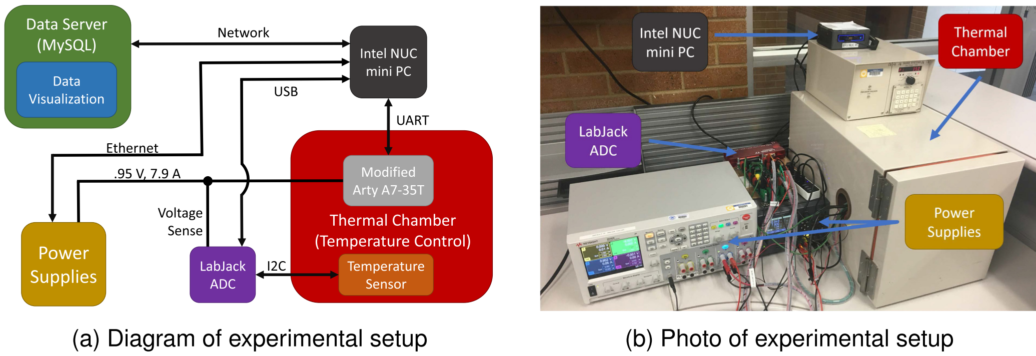


Fig. 11. Experimental Setup. The experiment is kept in a controlled environment by a thermal chamber and a high-precision power supply.

control voltage with a much higher degree of accuracy and also supply a higher amount of current than the on-board power regulator. The external power supply was configured to supply 0.95 V to the Arty board for internal logic and used sense lines to compensate for voltage drop over the supply lines.

The NUC PC polled the power supply for voltage and current values over Ethernet, and it would then log them to the database multiple times a second. As shown in the detailed results for the *High Density Bottom Experiment* (Table 1), this setup kept the voltage at $0.95 \pm 0.000\,52$ V for the duration of the experiment. In addition to the voltage and current reported by the power supply, we used a 0.1 Ω 0.05 % shunt resistor, monitored by the NUC via the LabJack, to verify the reported values.

Since our experiments ran for days or weeks at a time, we collected very large amounts of measurement data into our SQL database (several hundred gigabytes over months of experimentation). Significant effort was spent to manage this data and display it in an effective manner. This was managed by a custom Python package and series of Jupyter Notebooks, which queried our database, performed data processing, and used the HoloViews Python library to generate the various graphs and heatmaps presented in the next section. Additional scripting allowed us to annotate these graphs with different information from the FPGA floorplan.

5.3 High Density Bottom Experiment Results

The results of the *High Density Bottom Experiment* are summarized in Table 1, and the slowdown of the ROs is plotted in Figure 12.

5.3.1 Slowdown. Figure 12 shows the percent slowdown of frequency over time for each of the three distinct ROs. There are three periods shown in the plot: the first burn period, the recovery period, and the second burn period, which last 864, 624, and 2,208 hours respectively.

As can be seen, the bottom RO (yellow box in Figure 3) experiences a slowdown of 4.8% during the first burn period, which increases to 8.1% after the second burn period. The Middle RO (red) experiences a slowdown of 5.1%, which increases to a total of 8.5%. The top RO (blue) experiences a 1.4% slowdown after the initial burn, which increases to 2.1% after the second burn.

It can also be seen in Figure 12 that the degradation effects are heavily weighted towards the beginning of the burn period. After the first day, the Middle RO experiences a 1.25% decrease in frequency and by the end of the sixth day the slowdown reaches 2.56%, half of the total slowdown seen during the first burn period. This early drop-off is also seen in previous FPGA aging work [2, 5, 8, 10, 22]. In addition, while the slowdown during the first burn period resembles a decaying exponential, the slope of the slowdown becomes approximately linear by the 300-hour mark.

Table 1. Detailed Results for the *Initial Experiment*, Including Environmental Data

Time	Top-RO Slowdown	Mid-RO Slowdown	Bottom-RO Slowdown	I_{in} *During Burn*	V_{in} *During Characterize*	$T_{ambient}$ *During Characterize*
First Burn Period						
0 Days	−0.0% ± 0.02%	−0.0% ± 0.02%	−0.0% ± 0.02%	N/A	N/A	N/A
5 Days	−0.8% ± 0.03%	−2.4% ± 0.02%	−2.4% ± 0.02%	8,015 ± 9mA	952.2 ± 0.52mV	34.86 ± 0.03 °C
10 Days	−1.0% ± 0.03%	−3.1% ± 0.02%	−3.0% ± 0.02%	8,012 ± 10mA	952.3 ± 0.52mV	34.93 ± 0.04 °C
15 Days	−1.1% ± 0.02%	−3.7% ± 0.02%	−3.5% ± 0.02%	8,004 ± 9mA	952.3 ± 0.52mV	34.94 ± 0.06 °C
20 Days	−1.2% ± 0.02%	−4.2% ± 0.02%	−3.9% ± 0.02%	7,993 ± 9mA	952.3 ± 0.52mV	34.94 ± 0.04 °C
25 Days	−1.2% ± 0.03%	−4.5% ± 0.02%	−4.2% ± 0.02%	7,995 ± 11mA	952.3 ± 0.52mV	34.97 ± 0.06 °C
30 Days	−1.3% ± 0.02%	−4.9% ± 0.02%	−4.6% ± 0.02%	7,985 ± 11mA	952.3 ± 0.52mV	34.93 ± 0.03 °C
35 Days	−1.4% ± 0.03%	−5.1% ± 0.02%	−4.8% ± 0.02%	7,974 ± 10mA	952.3 ± 0.52mV	34.96 ± 0.05 °C
Recovery Period						
0 Days	−1.4% ± 0.02%	−5.1% ± 0.02%	−4.8% ± 0.02%	47 ± 1mA	952.3 ± 0.52mV	34.94 ± 0.07 °C
5 Days	−1.4% ± 0.02%	−5.1% ± 0.02%	−4.8% ± 0.02%	47 ± 1mA	952.3 ± 0.52mV	34.94 ± 0.07 °C
10 Days	−1.4% ± 0.02%	−5.2% ± 0.02%	−4.8% ± 0.02%	47 ± 1mA	952.3 ± 0.52mV	34.94 ± 0.06 °C
15 Days	−1.4% ± 0.02%	−5.2% ± 0.02%	−4.8% ± 0.02%	47 ± 1mA	952.3 ± 0.52mV	34.94 ± 0.03 °C
20 Days	−1.4% ± 0.02%	−5.2% ± 0.02%	−4.8% ± 0.02%	47 ± 1mA	952.3 ± 0.52mV	34.90 ± 0.04 °C
25 Days	−1.4% ± 0.02%	−5.2% ± 0.02%	−4.8% ± 0.02%	47 ± 1mA	952.3 ± 0.52mV	34.92 ± 0.05 °C
Second Burn Period						
0–21 Days	N/A	N/A	N/A	N/A	N/A	N/A
21 Days	−1.5% ± 0.02%	−6.1% ± 0.02%	−5.7% ± 0.05%	7,951 ± 10mA	N/A	35.12 ± 0.04 °C
33 Days	−1.7% ± 0.02%	−6.7% ± 0.02%	−6.2% ± 0.05%	7,939 ± 11mA	952.2 ± 0.52mV	35.13 ± 0.04 °C
45 Days	−1.8% ± 0.02%	−7.2% ± 0.02%	−6.7% ± 0.05%	7,948 ± 9mA	952.2 ± 0.52mV	35.01 ± 0.09 °C
57 Days	−1.8% ± 0.02%	−7.5% ± 0.02%	−7.1% ± 0.05%	7,946 ± 10mA	N/A	35.16 ± 0.04 °C
69 Days	−1.9% ± 0.02%	−7.9% ± 0.02%	−7.5% ± 0.04%	7,929 ± 10mA	952.2 ± 0.52mV	35.01 ± 0.05 °C
81 Days	−2.0% ± 0.02%	−8.2% ± 0.02%	−7.8% ± 0.04%	7,922 ± 11mA	952.2 ± 0.52mV	35.01 ± 0.04 °C
93 Days	−2.1% ± 0.02%	−8.5% ± 0.02%	−8.1% ± 0.04%	7,918 ± 15mA	952.3 ± 0.52mV	35.03 ± 0.04 °C
94 Days	−2.1% ± 0.02%	−8.5% ± 0.01%	−8.1% ± 0.04%	7,919 ± 15mA	952.3 ± 0.52mV	35.06 ± 0.04 °C

Note that there were some issues with the monitoring equipment during the beginning of the second burn period where we were able to burn but unable to collect accurate RO frequencies and environmental data.

5.3.2 Recovery. The total percent change for the Top, Middle, and Bottom ROs during the recovery period are −0.0090%, −0.0323%, and +0.0006%, respectively, which indicates that no significant recovery occurs during the recovery period. This can also be seen in Figure 12, which shows that the curves for each of the ROs completely flatten during the recovery period.

5.3.3 Temperature. During the burn period, the steady state temperature of the chip, according to the on-chip temperature sensor, is 177.7°C, significantly higher than the maximum temperature rating of 100°C for this device [13]. Furthermore, due to the heat spreading capabilities of ICs and the distance between the on-board temperature sensor and the transistors in the burn region, this value only represents a lower bound of the temperature actually experienced; the actual junction temperature seen by the transistors in the burn region is likely to be higher.

5.3.4 Current. In addition to RO frequency, the current observed within the burn period also decreases over the duration of the experiment. This can be seen in both Table 1 and Figure 13, which contains plots of the burn current during both burn periods. Figure 13(a) shows the current steadily decreasing over time, with a total decrease of 0.89% by the end of the first burn period. There is is also a sudden decrease of current that occurs at around 800 hours.

While Figure 13(b) shows an additional 0.69% decrease in current over the duration of the second burn period, the decrease is not as steady as in the first burn period. Instead, a large decrease in

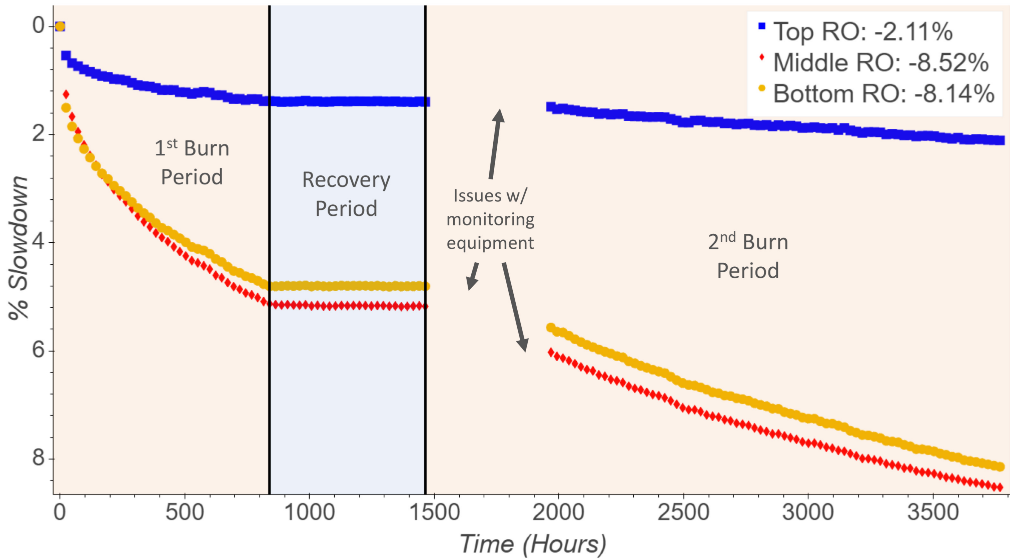
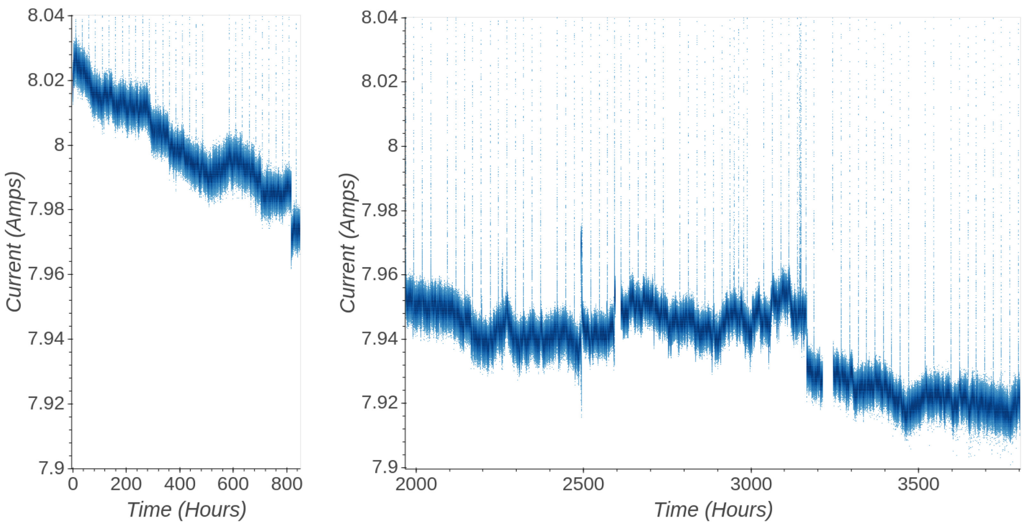


Fig. 12. Plot showing the percent slowdown of the *High Density Bottom Experiment's* three ROs over time. It includes three periods: a first and second burn period with an intermediate recovery period. Note that there were some issues with the monitoring equipment during the beginning of the second burn period where we were able to burn but unable to collect accurate RO frequencies.



(a) Plot showing the current during the first burn phase of the *High Density Bottom Experiment*.

(b) Plot showing the current during the second burn phase of the *High Density Bottom Experiment*. Note that this plot does not include the time the monitoring equipment was experiencing issues.

Fig. 13. Plots showing the current of the *High Density Bottom Experiment* during the first and second burn periods.

current happens suddenly around the 3,200-hour mark, which is similar to the sudden drop-off in current at the end of the first burn period.

Additionally, while not pictured in Figure 13, no noticeable change in current is seen during the recovery phase.

5.3.5 Verifying Functionality through Readback. We are not entirely sure what causes the decrease in current throughout both burn periods. Since temperature and voltage were kept within very precise ranges, we do not believe it is attributed to environmental changes. It seems most likely that the aging effects cause the threshold voltage of the transistors to increase, causing them to draw less current over time. We are also concerned, especially because of the few sudden drops in current, that perhaps electromigration is causing permanent failures in certain wire paths.

To test this possibility, we created designs that would test whether the LUT and FFs would all still transmit a logic-0 and logic-1. To do this, we created two designs. One design tests all of the shorted LUTs and the other tests all of the shorted FFs. Each design routes the LUT or FF used in our shorting experiments to an FF for capture. We use a custom-built router written with RapidWright to ensure the path from the shorted LUT or FF to the capture FF uses the same PIPs and wires that were used to create the short circuit.

To test for failures, we drive all of the LUTs or FFs from a logic-0 to a logic-1, perform a readback capture, and check to make sure that the value was successfully recorded by the capture FFs. We then do the same process again, this time driving a logic-1 to a logic-0. If any of the driven values are not recorded, then our experiment throws an error reporting the failed paths.

Using this framework, we were unable to detect any errors where a LUT or FF could not transmit a logic-0 or logic-1 along the routing used by the short circuits. This suggests that our aging technique never caused a permanent stuck-at fault throughout duration of the *High Density Bottom Experiment*.

5.4 High Density Top Experiment

5.4.1 Slowdown. Figure 14 shows a heatmap of percent slowdowns for the *High Density Top Experiment* after 30 days of burn. A three-dimensional topography of the same heatmap can be seen in Figure 15. The burn region is denoted by the green box in Figure 14. As can be seen in both figures, the highest concentration of damage occurs in the top-center of the chip, with the highest percent slowdown at 5.06%. The damage steadily decreases as the ROs move away from the top-center of the chip and continues to decrease all the way to the very bottom corners of the chip, as seen in Figure 15.

In addition to the heatmap, we chose three ROs—outlined in Figure 14—to show a more detailed analysis of the slowdown over time. These three ROs are in approximately inverted locations compared to the three distinct ROs used to characterize the *High Density Bottom Experiment*, and they allow us to make more direct comparisons between the *High Density Bottom* and *High Density Top* experiments.

A plot of the percent slowdown over time for these three ROs is presented in Figure 16. The ROs in this plot follow the same general trends as the three distinct ROs plotted in Figure 12. The frequency of the top (red) and middle (blue) ROs, which are both inside the burn region, decay at a much higher rate than the bottom (yellow) RO, which is located outside the burn region. Also, at the beginning of the burn period, the slope of the degradation of each of the ROs starts out as a decaying exponential, but becomes approximately linear within a few days of burn.

There are a couple of key observations from the *High Density Top Experiment* that are worth highlighting. The first is that the relative frequencies of the FPGA switch between the bottom and

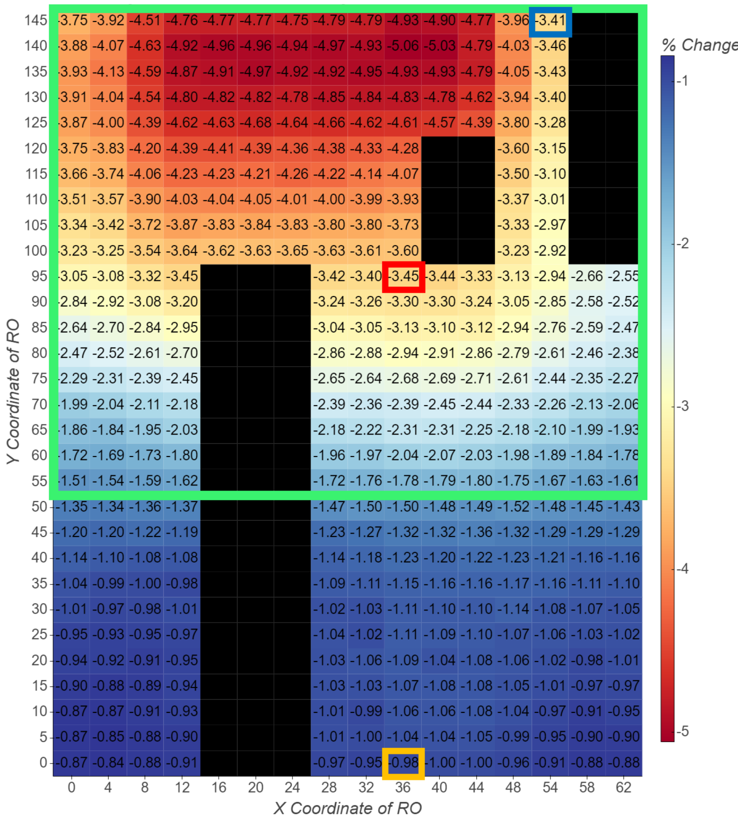


Fig. 14. Heatmap of the *High Density Top Experiment*, showing the percent slowdown after 30 days of burn. The green box shows the area of the chip that was shorted. The yellow, red, and blue boxes, respectively, show the bottom, middle, and top ROs, which can be seen in more detail in Figure 16.

top of the chip. This effect is illustrated in Figure 17, which shows the raw frequencies of the chip before and after the burn period. As can be seen, before any burn occurs, the relative frequency of the chip is generally higher at the top of the FPGA. After 30 burn days, the relative frequencies flip so the higher frequencies are found at the bottom of the chip. This shows that it is possible to alter the relative frequencies of ROs running on FPGA fabric by using a large area of shorts.

The second observation is the difference in the slowdown amount between the the *High Density Bottom* and *High Density Top* experiments. After 720 hours (30 days) of burn, the bottom, middle, and top ROs of the *High Density Top Experiment* experience a slowdown of 0.98%, 3.45%, and 3.41%, respectively. This is compared to the 1.34%, 4.81%, and 4.52% slowdown that the top, middle, and bottom ROs, respectively, experience after 720 hours of burn with the *High Density Bottom Experiment*. We are not sure why the *High Density Top Experiment* slows down slightly less than the *High Density Bottom Experiment*. This may be due to where the three ROs are located relative to where the most aged area is. As can be seen in Figure 14, none of the three ROs are located in the area that experienced the most degradation. Unfortunately, we do not have detailed heatmap data from the *High Density Bottom Experiment* to know how those three ROs behaved relative to their surroundings. It is also possible the difference is partly due to natural process variation between the FPGAs.

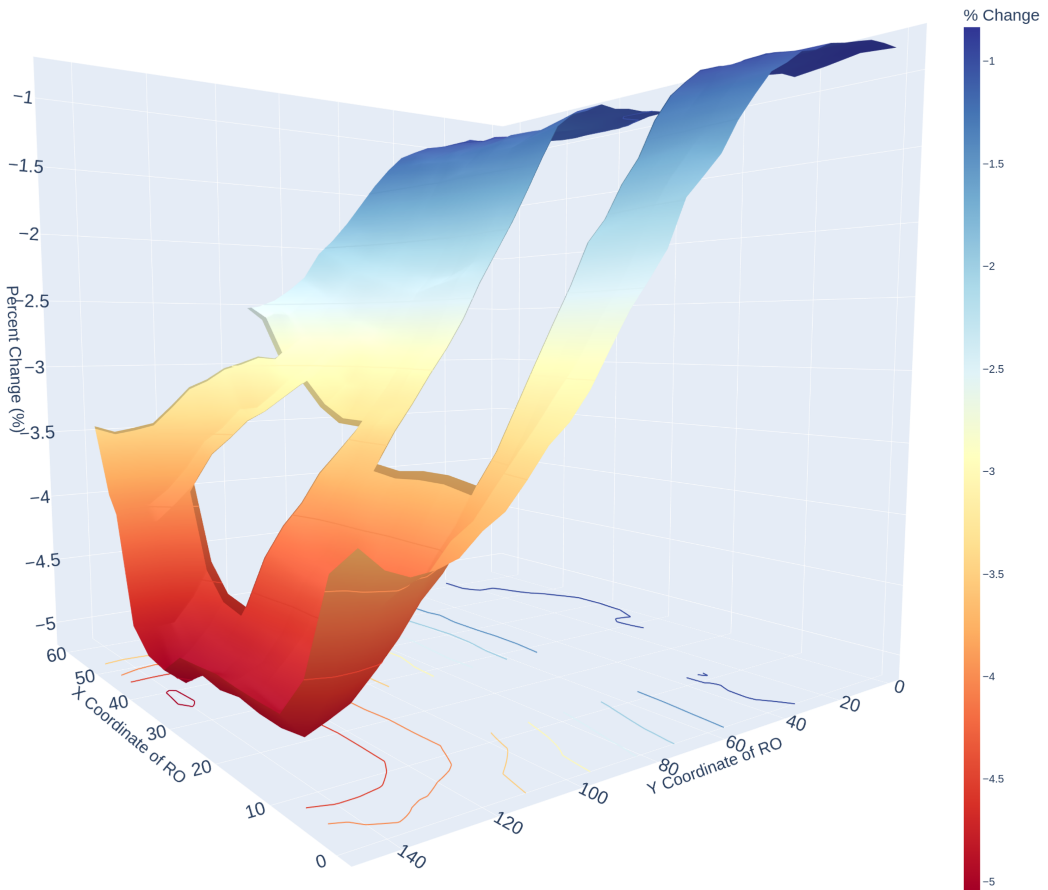


Fig. 15. A three-dimensional heatmap of the *High Density Top Experiment*, showing the percent slowdown after 30 days of burn. This heatmap emphasizes the slope of the damage caused by the short circuits. Along the y-axis, the slope is fairly steep inside the shorted region and starts to flatten out near the non-shorted region. Figure 14, only in three-dimensional form.

5.4.2 Recovery. Figure 16 also shows the 720-hour (30-day) recovery period we performed for this experiment. For all ROs, the largest recovery experienced during the recovery period was +0.029%, showing that no significant recovery took place during that time. This matches the results we observed for the *High Density Bottom Experiment* discussed in Section 5.3.2.

5.5 Checkerboard Experiment

5.5.1 Slowdown. Figure 18 shows a heatmap of percent slowdowns for the *Checkerboard Experiment* after 50 days of burn. The shorted region is denoted by the solid green lines containing the bolded numbers within. The highest concentration of damage is seen in the lower-right region of the chip, which had a maximum slowdown of 1.63%. This region of the chip is surrounded by shorts on three sides: above, below, and to the right, making it the highest concentration of shorts on the chip. In contrast, the shorted area in the middle-left side of the chip is isolated from the other shorted regions, and thus experiences the smallest slowdown of all the shorted areas on the chip.

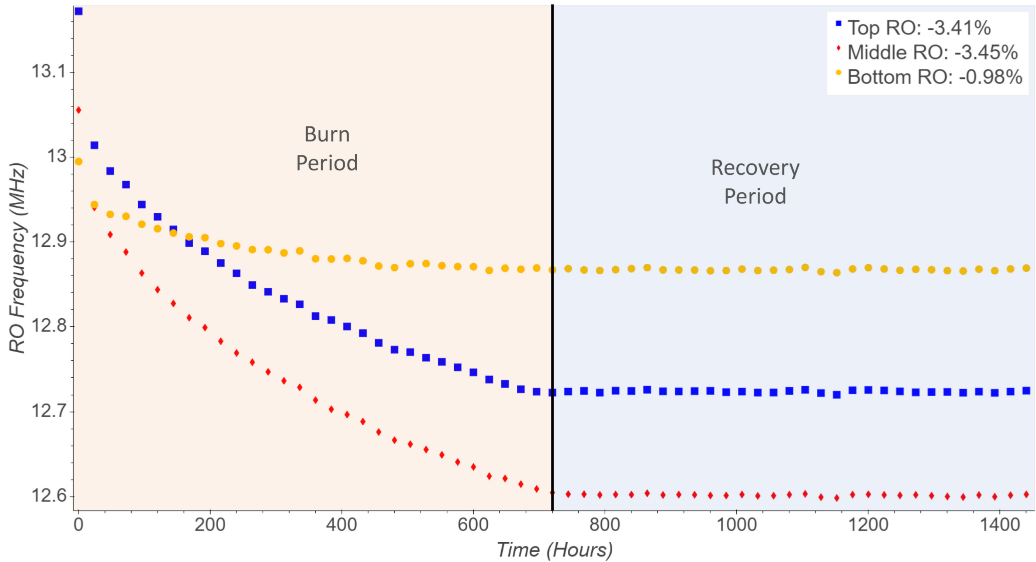
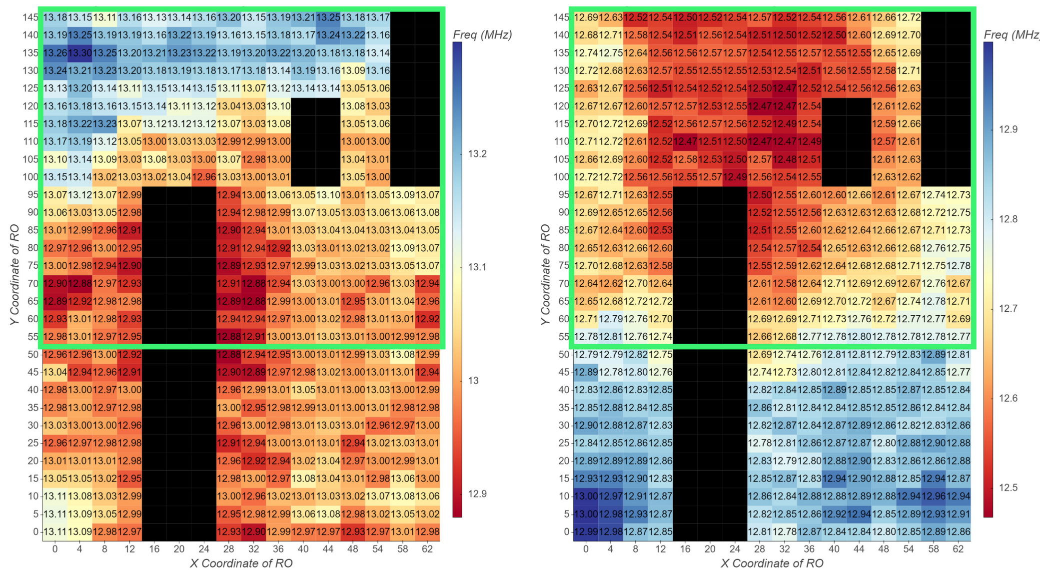


Fig. 16. Plot showing the percent slowdown of three ROs used to characterize the *High Density Top Experiment*.



(a) Heatmap showing the frequencies of ROs before any burn had occurred.

(b) Heatmap showing the frequencies of ROs after 30 days of burn had occurred.

Fig. 17. Two heatmaps of the *High Density Top Experiment*, showing the relative frequency flipping after 30 days of burn.

This difference in slowdown between the various regions of the chip indicates that the more shorts that are concentrated in a particular region, the more slowdown that region will experience. Further research needs to be done to determine exactly how the mass and geometry affect the slowdown of a shorted region.

Fig. 18. Heatmap of the *Checkerboard Experiment*, showing the percent slowdown after 50 days of burn. The green lines surrounding the bold regions show the area of the chip that are shorted during the experiment. The blue lines surround the area of the chip that experienced the most slowdown. The average slowdown of this area over time is shown in Figure 19.

Another interesting characteristic seen in this experiment is that high-level damage "leaks" out into an non-shorted region of the chip. This can be seen in the lower right region where a non-shorted area directly left of the concentrated shorted region experiences up to 1.36% slowdown, only 0.27% lower than the maximum damage seen in the nearby shorted region, and much higher than the damage seen in most other shorted areas. This may be due to heat being the primary cause of the slowdown experienced by our ROs, as the heat caused by short circuits is able to spread outside of the shorted regions.

The average percent slowdown over time of the most degraded region is plotted in Figure 19. This region is boxed in blue in Figure 18. The slowdown during the burn period follows the same general trend as the two high-density experiments. The degradation starts off exponential, but quickly becomes linear within the first few days.

5.5.2 Recovery. For this experiment, we performed a recovery phase lasting 720 hours (30 days) and can be seen in Figure 18. Despite this recovery period experiencing more noise than the two high-density experiments, the maximum recovery during this phase of the *Checkerboard*

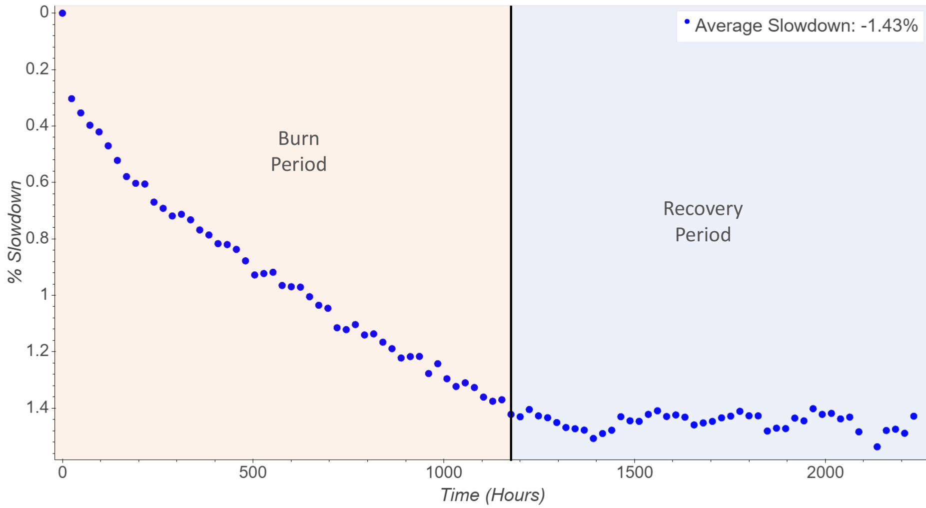


Fig. 19. The average slowdown over time for a particular region of the device in the *Checkerboard* experiment. This region is boxed in blue in Figure 18.

Experiment was only 0.0082%. This corresponds to the *Initial* and *High Density Top Experiments* in that no significant recovery occurred.

5.6 Inverse Checkerboard

5.6.1 Slowdown. Figure 20 provides a heatmap detailing the percent slowdown for the *Inverse Checkerboard Experiment* after eight days of burn. The experiment was short in length due to the FPGA board failing after eight days of burn had occurred. While the cause of the failure is ultimately unknown, it was likely due to an error in the board modification process, as opposed to the stress induced by running short circuits. Despite the failure, most of the damage caused by short circuits typically takes place within the first few days of burn, thus the results of this experiment are still comparable to the *Checkerboard Experiment*.

The shorted burn regions are denoted in Figure 20 by the solid green lines containing the bolded numbers within. In general, the highest slowdown occurs in the largest shorted regions, which are at the top and bottom of the chip. In the rectangular regions on the bottom-left and top-left areas of the FPGA, we see maximum percent slowdowns of 0.93% and 0.91%, respectively. When we average the slowdowns of the individual ROs in the bottom-left and top-left shorted regions, we see a mean slowdown of 0.74% and 0.72%, respectively.

The two small regions on the right side of the chip see less of a slowdown, with a combined maximum slowdown of 0.62% and a combined mean slowdown of 0.60%. These two regions experience only slightly more slowdown (<0.1%) than the surrounding non-shorted regions. This matches with our observations from the *Checkerboard Experiment* in Section 5.5 that larger shorted regions have significantly more slowdown than smaller shorted regions and further supports the theory that heat is the primary cause of slowdown experienced by our ROs. Furthermore, this result suggests that creating even smaller regions of short circuits will likely not provide much impact for localized aging.

Similar to the *Checkerboard Experiment*, the *Inverse Checkerboard Experiment* also sees high-level damage “leak” from certain shorted regions. The non-shorted regions directly to the right of both the top and bottom shorted regions see a slowdown that is similar to the slowdown seen inside

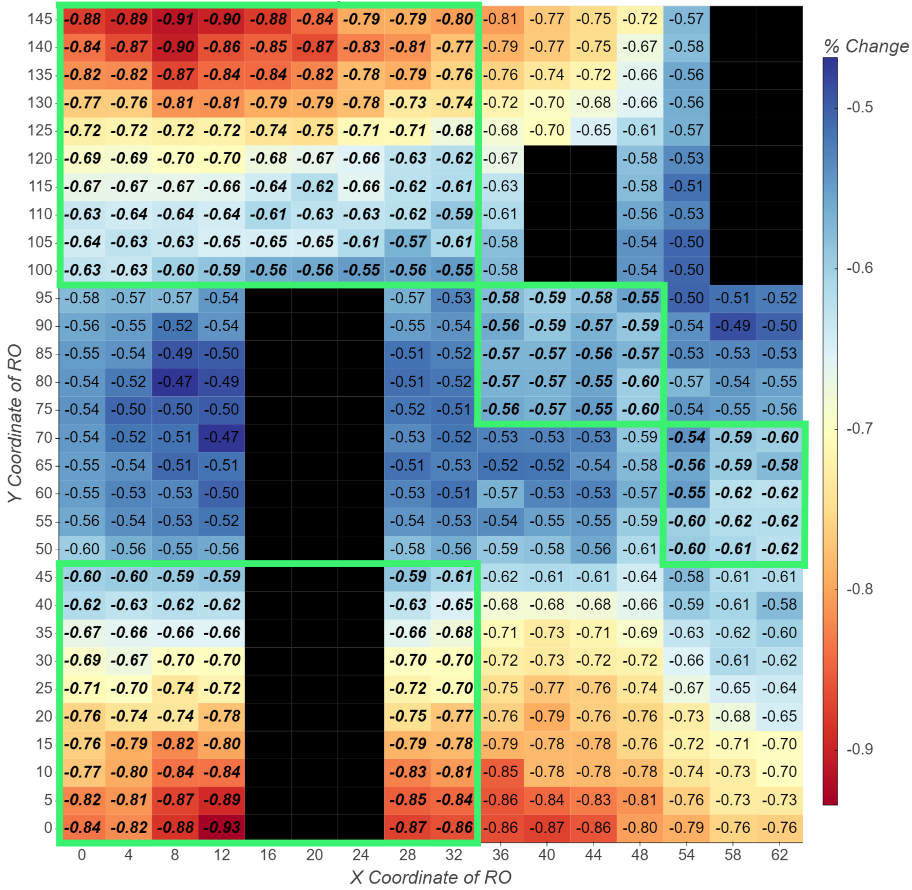


Fig. 20. Heatmap of the *Inverse Checkerboard Experiment*, showing the percent slowdown after eight days of burn. The solid green lines surrounding the bold regions show the area of the chip that was shorted.

those shorted regions. For the top of the chip, the maximum slowdown seen outside the shorted region is 0.81%, which is 0.10% lower than the maximum slowdown seen inside the top shorted region. For the bottom, the maximum slowdown is 0.87% outside the shorted region, which is only 0.06% lower than the maximum slowdown seen inside the bottom shorted region. Once again, this is likely due to heat being the primary cause of slowdown experienced by our shorts, as heat can easily spread outside shorted regions.

Additionally, the slowdown seems to be concentrated to both the very top and bottom parts of the chip. This is similar to what was observed in the *High Density Top Experiment* where slowdown within the shorted regions was concentrated towards the upper part of the chip. Assuming that heat is the primary cause of slowdown, the high level of slowdown at the top and bottom of the chip may be contributed to the heat-spreading characteristics of the packaging housing the FPGA.

5.6.2 Recovery. Unfortunately, since the board for the *Inverse Checkerboard Experiment* malfunctioned unexpectedly, we were unable to perform recovery testing on this part.

Table 2. Comparison of Our Experiments with Previous Works

Experiment	Stress Duration	Methodology	Max % Slowdown
High Density Bottom	129 Days	20,798 shorts on the bottom two-thirds of the board	8.51%
High Density Top	30 Days	20,798 shorts on the top two-thirds of the board	5.06%
Checkerboard	50 Days	17,600 shorts laid out in a checkerboard pattern	1.62%
Reverse Checkerboard	8 Days	14,900 shorts laid out in a checkerboard pattern	0.93%
Stott et al. [10]	75 Days	Increase temperature and voltage to 147 °C and 1.5 V	15%
Maiti et al. [5]	17 Days	Increase temperature and voltage to 70–80 °C and 1.5–1.8 V	6.7%
Slimani et al. [22]	14 Days	Increase temperature to 120 °C	1.8%

6 DISCUSSION

This section discusses the results of our experiments and the several implications derived from them. These include different variables that affect the slowdown rate, the locality of the degradation caused by short circuits, the lack of recovery seen in our experiments, the potential causes of degradation, and the aging mechanisms involved with the shorts.

6.1 Comparison of Experiments

A summary of the comparison between our experiments, as well as the experiments of previous work discussed in Section 2.2, are summarized in Table 2. This table shows that while our method has comparable results to those of previous work, it is not quite as effective as the methods proposed by Stott et al. and Maiti et al., which is likely due to them raising the voltage in addition to raising the temperature [5, 10].

6.2 Slowdown Rate

One variable that affects the slowdown rate is the duration of a burn. While all four experiments show that the slowdown during a burn period is initially a decaying exponential, with the largest slowdown occurring during the beginning of the burn period, the second burn period of the *High Density Bottom Experiment* shows that the slowdown rate quickly becomes linear with a steady slope that experiences little change even after months of burn. By measuring this slope, slowdown caused by short circuits become reliably predictable after just a few weeks of burn.

The two checkerboard experiments show that the concentration of shorts in a burn region also affects the rate of slowdown. While the *High Density Bottom Experiment* and *High Density Top Experiment* were able to achieve a maximum slowdown of 2.87% and 2.15%, respectively, after eight days of burn, the *Checkerboard Experiment* and *Inverse Checkerboard Experiment* achieved a maximum slowdown of only 0.75% and 0.93%, respectively, after eight days of burn. This is due to the two checkerboard experiments having much smaller burn regions than the regions used in the two high-density experiments.

Last, different locations experience different slowdown rates. Even within the same burn region, the *High Density Top Experiment* shows ROs with vastly different slowdowns, with a 3.55% difference in slowdown between the maximum and minimum slowdowns within the burn region. This experiment suggests that slowdown tends to occur most at the top of the chip, while the *Inverse*

Checkerboard Experiment shows that both the top and bottom of the chip are most affected by burn regions. This is potentially due to the heat-spreading characteristics of the packaging.

6.3 Aging Locality

While all experiments show that shorted regions generally age more than non-shorted regions, achieving a precisely targeted localized slowdown is challenging. For example, the *High Density Top Experiment* shows that while a significant localized slowdown does occur with a large number of short circuits, the slowdown varies widely across the shorted region.

The checkerboard experiments show that using smaller burn regions to try to achieve a precisely targeted slowdown has two drawbacks. The first is that there is significantly less of a difference between slowdown inside of the burn regions and slowdown outside of the burn regions. For the *Inverse Checkerboard Experiment*, the difference between the minimum slowdown and the maximum slowdown is just 0.46% after nine days of burn, compared to the 2.54% difference in the *High Density Top Experiment* after the same amount of burn.

The second drawback is that slowdown tends to "leak" outside of the shorted region. Both checkerboard experiments show that some areas just outside the burn region show almost as much slowdown as the maximum slowdown seen inside the chip. For example, the *Inverse Checkerboard Experiment* sees a maximum slowdown of 0.93% inside the burn regions, and 0.87% directly outside, which is a difference of only 0.06%.

While short circuits cannot provide precisely targeted slowdown, our experiments still show that there is a significant difference between overall slowdowns inside and outside of a burn region. This shows that generally targeted localized slowdowns are still possible with our technique. In particular, Figure 17 shows that it is possible to flip the relative frequencies between the top and bottom of the chip.

6.4 Recovery

The *High Density Top Experiment* saw the most recovery out of all of our experiments. Despite this, the maximum recovery seen in the *High Density Top Experiment* was only +0.029%, which is well within the noise range of our measurements. Since none of our experiments exhibited significant recovery, it appears that the aging caused by short circuits is relatively permanent.

6.5 Aging Cause

As previously mentioned, we believe that heat, rather than current, is the primary cause of the slowdown seen by our ROs. This is supported by several observations made throughout our experiments.

The first is that the slowdown inside a shorted region varies widely, despite the configuration of the short circuits being the same throughout. For example, the maximum slowdown for the *High Density Top Experiment* is 5.06%, while the minimum is just 1.51%. This large difference between slowdowns inside the same shorted region is likely caused by heat, which can vary throughout the chip due to the heat-spreading capabilities of the physical layout and packaging.

Other work shows that self-heating caused by designs on an FPGA can lead to a 8 °C temperature difference on different locations of the device [28, 29]. While this is not a substantial difference across the chip, in this work the authors had only five ROs running on the chip, so the produced heat would likely be much lower than our experiments. In our case the total heat produced by the thousands of short circuits may lead to an increased temperature gradient. We attempted to perform a crude measurement of the heat gradient across the chip using an IR camera. This is shown in Figure 21, which provides a top-down view of the chip with the *High Density Top* configuration

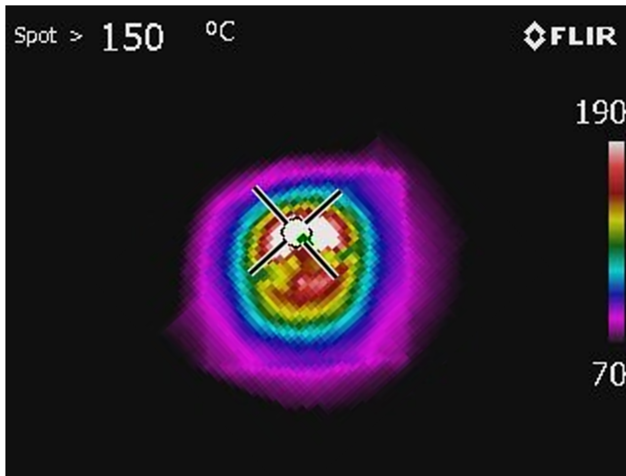


Fig. 21. A picture of an XC7A35T FPGA taken by a FLIR E8 thermal camera while the *High Density Top* shorts configuration is running. The heat is concentrated at the top of the chip, where the short circuits are placed.

loaded. It can be seen that the heat is concentrated at the top of the chip, which is where the short circuits are placed.

The slowdown “leakage” that occurs in both checkerboard experiments also supports heat being the contributing factor. In these cases, the slowdown seen outside the shorted region is similar to the maximum slowdown seen inside the shorted regions.

Although we believe heat to be the primary cause of the observed slowdown, it is still very possible that additional aging is being caused by the high levels of current. It is likely that high levels of current are causing degradation in the form of HCI and EM; however, our characterization technique is unable to distinguish this type of degradation from the degradation caused by heat. Further work must be done to be able to isolate damage caused by current from damage caused by heat.

In addition, we do not know the exact transistor layout of the parts, so if the current produced by our short circuits is indeed damaging particular transistors more than others, we are not sure whether or not our ROs would use the same transistors and would thus accurately detect this damage. We believe it may be possible with more deliberate RO routing strategies to ensure that the PIPs used for our ROs match the PIPs used for short circuits. For the experiments presented in this article, however, we were unable to do this. We hope to do so in future work.

6.6 Aging Mechanisms

As first discussed in Section 2.1, there are several established aging mechanisms that are possible contributors to the FPGA fabric’s degradation. While we are unable to determine the exact mechanisms at play, there is evidence to suggest that our shorts cause some level of BTI, EM, and HCI effects to occur.

6.6.1 Bias Temperature Instability. BTI, as presented in Section 2.1.1, becomes more pronounced in the presence of increased temperature and increased voltage. While we demonstrate in Section 5.2 that no external factor could be causing accelerated BTI degradation, in Section 5.3.3, we show that the short circuits cause on-chip temperatures to exceed 170 °C during the burn phase of the *High Density Bottom Experiment.* In Reference [22] it is shown that the temperature only needs to be raised to 125 °C for the Artix 7 part to experience pronounced NBTI effects. As such,

we feel that accelerated BTI is a major contributor of the accelerated aging we observed in our experiments.

This claim is further supported by the pronounced initial slowdown. Similar results, where the initial degradation is the most pronounced, are reported in References [2, 5, 8, 10, 22], all of which examine the effects of NBTI on FPGAs.

Additionally, since the main accelerator of BTI is heat, damage from BTI is not limited to only affecting shorted transistors. This makes it much easier to detect by our ROs, as BTI can affect all transistors in a RO, as opposed to just the shorted ones. BTI also has the potential to spread outside of a shorted region, which is what is observed in all of our experiments.

One defining characteristic of BTI is its ability to recover [30, 31]. While such recovery has been seen in other studies that use NBTI to age FPGAs [5, 22], we do not see recovery in any of our experiments. This contradicts our belief that BTI is a major factor of the accelerated aging we observe in our experiments. Further research must be done to explore why our experiments experience no recovery despite the evidence that BTI is occurring.

6.6.2 Electromigration. EM, as presented in Section 2.1.2, becomes more pronounced in the presence of increased temperature and high DC current. In our *High Density Bottom Experiment*, we have shown currents in excess of 7.9 A—or 0.38 μA per short—and temperatures in excess of 170 °C. These environmental conditions should be sufficient to accelerate EM, leading us to believe that accelerated EM does occur during the burn phases of our experiments.

Unfortunately, we are unable to determine if EM contributed to any of the slowdown seen by our ROs. This may be due to the effects of EM being local to only the shorted transistors, or possibly because the slowdown caused by the thinning of a few wires in the routing is negligible.

In either case, the concern with EM is more about wire failure than a degradation of switching speeds. Through readback, we are able to determine that EM has not caused any routes to fail for the *High Density Bottom Experiment*. This suggests that the interconnects inside the FPGA used in the *High Density Bottom Experiment* are fairly resistant to EM.

6.6.3 Hot Carrier Injection. HCI, as presented in Section 2.1.3, is accelerated through an increase in the substrate current of a transistor. The short circuits used in our experiments greatly increase the drain current of select transistors, which should also increase the substrate current of those same transistors to some extent [32, 33] and accelerate HCI.

HCI is similar to EM in that it is a localized mechanism that only affects the shorted transistors in our experiments. So, while we do believe that HCI is being accelerated by our short circuits, our ROs may be unable to detect it.

6.6.4 Time-Dependent Dielectric Breakdown. TDDB, as presented in Section 2.1.4, is accelerated through increased voltage and decreased temperature. Since all of our experiments were conducted at nominal voltages and increased temperatures, we do not believe that our short circuits accelerated TDDB.

6.7 Process Variation, Device Lifespan, and Timing Guardband

Ideally, we would have liked to demonstrate our aging technique on many different FPGA boards to demonstrate its effectiveness in the presence of different process variations that may exist across different boards. In the process of characterizing each board, we did observe that they had measurably different starting frequencies. Figure 22 shows the variation for the initial frequency of the boards used in three of our experiments. This shows that the average RO frequency across the boards used in this work ranged from 13.05 MHz to 13.37 MHz, a 2.5% difference. Since this work,

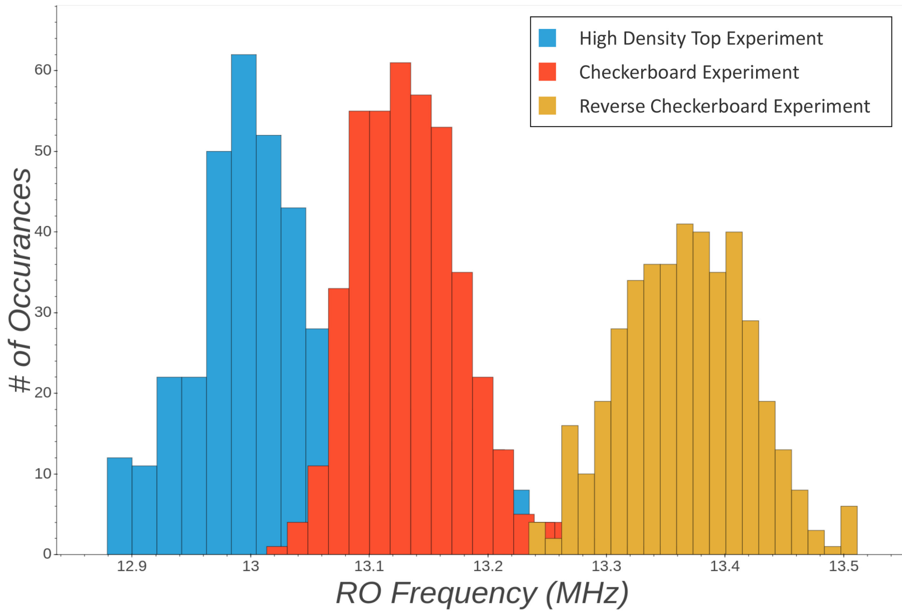


Fig. 22. The initial raw frequencies of the *High Density Top*, *Checkerboard*, and *Reverse Checkerboard* experiments. These frequencies were observed at the beginning of their respective experiments, before any were exposed to short circuits. The variation between the *High Density Top* and the *Reverse Checkerboard* experiments is 2.5%.

we have characterized seven additional non-aged boards. These ROs on these boards have average frequencies that range from 12.85 MHz to 13.92 MHz, a 7.7% difference.

While we do not observe an obvious difference in aging behavior between the different boards, it would be interesting future work to perform a detailed study as to whether the starting frequency of a board affects the rate at which it experiences aging.

Due to **process, voltage, and temperature (PVT)** variation of CMOS technology, as well as frequency degradation caused by transistor aging, FPGA vendors include a guardband when reporting timing delay in their tools. Yu et al. has found that in Xilinx's Spartan-3e family this guardband is on average 10%, meaning that the wire delay reported by the Xilinx tools is on average 10% higher than the actual wire delay of the device [34].

Section 4.3.1 shows that the *High Density Bottom* experiment experiences a 8.5% degradation caused by our shorted configuration. This is approaching the 10% guardband from Reference [34]. While that work targeted a different FPGA family, so it is not an ideal comparison, it suggests we have aged the FPGA to near the end of its lifetime. It also leaves only a small margin to account for PVT variations. Such a degradation is severe enough that timing analyses performed by Xilinx tools may no longer be accurate for certain chips running at higher temperatures and lower voltages. This could lead to timing violations for designs running on an aged FPGA and would be an interesting area for future work.

Additionally, since the degradation over time is shown to be linear at the end of the *High Density Bottom* experiment, it is likely that further aging would overcome the 10% guardband, further increasing the chance of a timing violation.

Ideally, we would like to correlate the aging caused by our short circuits to some number of years of aging caused by a real design. However, we have been unable to locate previous studies that

have documented the aging profile of these chips over years of their lifespan or find an accurate technique to predict this. One method is to calculate the aging acceleration factor, as discussed in Reference [8]. However, this predictive technique requires the actual junction temperature of the transistors and, since we are self-heating the chip (rather than using an external heat source), we are not confident that the 177.7 °C reported from *sysmon* is accurate of the actual junction temperature.

6.8 Security Implications

Short circuit–induced aging has a number of security implications for FPGAs. As discussed in the previous section, the aging may be sufficient to cause timing violations and break the functionality of designs. This could be exploited by an attacker to possibly perform some type of denial of service attack. This particular aging attack has the advantage to the attacker in that they do not need physical access to the device. They do, however, need to have the ability to program an arbitrary bitstream onto the device. The device must also have a power supply capable of handling high currents. Once these two criteria are met, the attacker is able to configure the FPGA with short circuits. If this configuration runs long enough, then the intrinsic speed of the chip will degrade past the guardband, causing critical paths to fail. Furthermore, since the aging is non-uniform, if the attacker has knowledge of the design that will be used on the chip in the future, then it may even be possible to perform a more targeted attack, causing timing failures in particular modules of a design.

Another security implication is using short circuit–induced aging to detect recycled FPGAs. Dogan et al. present a method to detect recycled FPGAs by looking at the aging curve of the chip. One disadvantage of this method is that the whole chip must undergo accelerated aging to accurately determine whether it is recycled. In Dogan et al., the FPGAs experience up to a 1.49% degradation before they perform the detection test [4]. Instead of aging the entire device, the detection method could be done using a relatively small shorted region, similar to the regions seen in the checkerboard experiment. This may enable the detection of recycled FPGAs with only a small portion of the chip experiencing significant degradation.

Short circuit–induced aging may also be used to create a watermark for an FPGA. Different patterns of short circuits could be placed throughout the chip, causing different areas of the chip to age at various rates. The difference in aging could then be detected with a heatmap characterization technique and would allow for identifying an watermarked FPGA or a group of FPGAs solely by the characterization.

7 CONCLUSION

This article introduces a new technique for accelerated aging on FPGAs and demonstrates how short circuits can induce said aging on an FPGA. By configuring the FPGA with tens of thousands of short circuits, we generate an excessive amount of current and heat, well outside normal operating ranges. This aging technique is novel in that it is independent of environmental conditions, such as input voltage and ambient temperature. Using this technique, we are able to successfully slow down the Xilinx Artix-7 fabric by over 5% in 36 days. After several months of burning the same chip, we are able to induce a slowdown of over 8.5%, proving that we can create sustained, linear damage over a long period of time. Compared to previous techniques, this method appears to provide significant aging capability, which aging is shown to be permanent. Most interestingly, this new technique is shown to create nonuniform aging effects in the FPGA fabric, where the maximum slowdown outside the short circuit region exhibited approximately one-third of the slowdown of the maximum slowdown inside the short circuit region.

We are able to demonstrate repeatability in the presence of process variations across different boards. Through the *High Density Top Experiment*, we show that the damage induced is not simply due to the fabric of the FPGA, but rather due to the placement of short circuits on certain locations of the chip. Additionally, due to the implementation of a more robust heatmap characterization method, we are able to demonstrate some key observations about how short circuit–induced aging occurs on an FPGA and the causes behind it. Through our two checkerboard experiments, we observe a positive correlation between the concentration of shorts and the amount of degradation that occurred in any particular region.

We also observe that in both checkerboard experiments high levels of damage can "leak" out from higher concentrated shorted regions. Additionally, both the *Inverse Initial* and *Inverse Checkerboard* experiments show that the highest levels of damage tend to concentration on the top and bottom of the FPGA. These two phenomena support that short circuit produced heat, rather than current, is the primary cause of damage seen by our characterizations.

Finally, none of the three experiments that underwent a recovery period experience any recovery, indicating that the damage caused by our aging method is permanent.

These results demonstrate that using short circuits are a viable technique for accelerated aging on FPGAs. Not only are the results comparable to other common techniques, but short circuit–induced aging has a number of interesting applications due to the localized nature of the degradation. First, the results show that it is possible to target a specific area of the chip, meaning that with knowledge of the resident applications, one could potentially target specific parts of the design. As we discussed, it appears that our technique approaches, and could likely exceed, the timing guardband provided by CAD tools. This means it may be possible to attack and break certain parts of the FPGA design. In addition, there are other interested security applications that could stem from this work, such as remote aging attacks, detecting recycled FPGAs with less degradation (as in Reference [4]), and watermarking FGPAs. Our hope is that thoroughly outlining the characteristics of short circuit–induced aging will encourage others to use and explore short circuits in their own works.

REFERENCES

[1] S. Novak, C. Parker, D. Becher, M. Liu, M. Agostinelli, M. Chahal, P. Packan, P. Nayak, S. Ramey, and S. Natarajan. 2015. Transistor aging and reliability in 14nm tri-gate technology. In *Reliability Physics Symposium.* DOI: http://dx.doi.org/10.1109/IRPS.2015.7112692

[2] Abdulazim Amouri, Florent Bruguier, Saman Kiamehr, Pascal Benoit, Lionel Torres, and Mehdi Tahoori. 2014. Aging effects in FPGAs: An experimental analysis. In *Conference on Field Programmable Logic and Applications (FPL).* 1–4. DOI: http://dx.doi.org/10.1109/FPL.2014.6927390

[3] Zana Ghaderi and Eli Bozorgzadeh. 2016. Aging-aware high-level physical planning for reconfigurable systems. In *Asia and South Pacific Design Automation Conference (ASP-DAC).* 631–636. DOI: http://dx.doi.org/10.1109/ASPDAC.2016.7428082

[4] Halit Dogan, Domenic Forte, and Mark Mohammad Tehranipoor. 2014. Aging analysis for recycled FPGA detection. In *Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT).* 171–176. DOI: http://dx.doi.org/10.1109/DFT.2014.6962099

[5] Abhranil Maiti, Logan McDougall, and Patrick Schaumont. 2011. The impact of aging on an FPGA-based physical unclonable function. In *Conference on Field Programmable Logic and Applications (FPL).* 151–156. DOI: http://dx.doi.org/10.1109/FPL.2011.35

[6] N. H. E. Weste and D. Harris. 2005. CMOS VLSI Design A Circuits and Systems Perspective, MA, Reading:Addison-Wesley. http://swarm.cs.pub.ro/~mbarbulescu/SMPA/CMOS-VLSI-design.pdf.

[7] JEDEC Solid State Technology Association and others. JEP122C. ([n. d.]). https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.478.4344&rep=rep1&type=pdf.

[8] Stefan Gehrer, Sebastien Leger, and Georg Sigl. 2015. Aging effects on ring-oscillator-based physical unclonable functions on FPGAs. In *Conference on ReConFigurable Computing and FPGAs (ReConFig).* 1–6. DOI: http://dx.doi.org/10.1109/ReConFig.2015.7393289

[9] Stefan Gehrer. Highly efficient implementation of physical unclonable functions on FPGAs. The Technical University of Munich. https://mediatum.ub.tum.de/doc/1355406/1355406.pdf.

[10] Edward A. Stott, Justin S. J. Wong, Pete Sedcole, and Peter Y. K. Cheung. Degradation in FPGAs: Measurement and modelling. 2010. In *Symposium on Field Programmable Gate Arrays (FPGA'10)*. Association for Computing Machinery, 229–238. DOI: http://dx.doi.org/10.1145/1723112.1723152

[11] T. T. Kim, P. Lu, and C. H. Kim. 2012. Design of ring oscillator structures for measuring isolated NBTI and PBTI. In *IEEE International Symposium on Circuits and Systems (ISCAS)*. 1580–1583. DOI: http://dx.doi.org/10.1109/ISCAS.2012.6271555

[12] S. Zafar, Y. Kim, V. Narayanan, C. Cabral, V. Paruchuri, B. Doris, J. Stathis, A. Callegari, and M. Chudzik. 2006. A comparative study of NBTI and PBTI (charge trapping) in SiO2/HfO2 stacks with FUSI, TiN, Re Gates. In *Symposium on VLSI Technology*. 23–25. DOI: http://dx.doi.org/10.1109/VLSIT.2006.1705198

[13] Xilinx. Artix-7 FPGAs data sheet: DC and AC switching characteristics (DS181). ([n. d.]), 64. https://www.xilinx.com/support/documentation/data_sheets/ds181_Artix_7_Data_Sheet.pdf.

[14] Z. Sun, S. Sadiqbatcha, H. Zhao, and S. X. Tan. 2018. Accelerating electromigration aging for fast failure detection for nanometer ICs. In *23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*. 623–630. DOI: http://dx.doi.org/10.1109/ASPDAC.2018.8297392

[15] M. Naouss and F. Marc. 2016. FPGA LUT delay degradation due to HCI: Experiment and simulation results. *Microelectron. Reliab.* 64 (2016), 31–35. DOI: http://dx.doi.org/10.1016/j.microrel.2016.07.048

[16] X. Wang, Q. Tang, P. Jain, D. Jiao, and C. H. Kim. 2015. The dependence of BTI and HCI-induced frequency degradation on interconnect length and its circuit level implications. *IEEE Trans. Very Large Scale Integ. Syst.* 23, 2 (2015), 280–291. DOI: http://dx.doi.org/10.1109/TVLSI.2014.2307589

[17] J. Keane, X. Wang, D. Persaud, and C. H. Kim. 2010. An all-in-one silicon odometer for separately monitoring HCI, BTI, and TDDB. *IEEE J. Solid-state Circ.* 45, 4 (2010), 817–829. DOI: http://dx.doi.org/10.1109/JSSC.2010.2040125

[18] M. Arabi, X. Federspiel, F. Cacho, M. Rafik, A.-P. Nguyen, X. Garros, and G. Ghibaudo. 2019. Temperature dependence of TDDB at high frequency in 28FDSOI. *Microelectron. Reliab.* 100–101 (2019), 113422. DOI: http://dx.doi.org/10.1016/j.microrel.2019.113422

[19] Christian Beckhoff, Dirk Koch, and Jim Torresen. 2010. Short-circuits on FPGAs caused by partial runtime reconfiguration. In *Conference on Field Programmable Logic and Applications (FPL)*. 596–601. DOI: http://dx.doi.org/10.1109/FPL.2010.117

[20] Ilija Hadžić, Sanjay Udani, and Jonathan M. Smith. FPGA viruses. 1999. In *Conference on Field Programmable Logic and Applications (FPL) (Lecture Notes in Computer Science)*, Patrick Lysaght, James Irvine, and Reiner Hartenstein (Eds.). Springer, 291–300. DOI: http://dx.doi.org/10.1007/978-3-540-48302-1_30

[21] Brad L. Hutchings, Joshua Monson, Danny Savory, and Jared Keeley. 2014. A power side-channel-based digital to analog converter for Xilinx FPGAs. In *Symposium on Field-Programmable Gate Arrays (FPGA'14)*. Association for Computing Machinery, 113–116. DOI: http://dx.doi.org/10.1145/2554688.2554770

[22] M. Slimani, K. Benkalaia, and L. Naviner. 2017. Analysis of ageing effects on ARTIX7 XILINX FPGA. *Microelectronics Reliability* 76–77 (2017), 168–173.

[23] R. S. Chakraborty, I. Saha, A. Palchaudhuri, and G. K. Naik. 2013. Hardware trojan insertion by direct modification of FPGA configuration bitstream. *IEEE Design Test* 30, 2 (2013), 45–54.

[24] Kenneth M. Zick and John P. Hayes. 2012. Low-cost sensing with ring oscillator arrays for healthier reconfigurable systems. *ACM Trans. Reconfig. Technol. Syst.* 5, 1 (2012), 1–26.

[25] Eddie Hung, Joshua M. Levine, Edward Stott, George A. Constantinides, and Wayne Luk. 2015. Delay-bounded routing for shadow registers. In *ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*. 56–65.

[26] Chris Lavin and Alireza Kaviani. 2018. RapidWright: Enabling custom crafted implementations for FPGAs. In *Symposium on Field-programmable Custom Computing Machines (FCCM)*. 133–140. DOI: http://dx.doi.org/10.1109/FCCM.2018.00030

[27] T. Gaskin, H. Cook, W. Stirk, R. Lucas, J. Goeders, and B. Hutchings. 2020. Using novel configuration techniques for accelerated FPGA aging. In *30th International Conference on Field-Programmable Logic and Applications (FPL)*. 169–175. DOI: http://dx.doi.org/10.1109/FPL50879.2020.00037

[28] Carsten Tradowsky, Enrique Cordero, Thorsten Deuser, Michael Hübner, and Jürgen Becker. 2012. Determination of on-chip temperature gradients on reconfigurable hardware. In *International Conference on Reconfigurable Computing and FPGAs*. 1–8. DOI: http://dx.doi.org/10.1109/ReConFig.2012.6416738

[29] John J. León Franco, Eduardo Boemo, Encarnación Castillo, and Luis Parrilla. 2010. Ring oscillators as thermal sensors in FPGAs: Experiments in low voltage. In *Southern Programmable Logic Conference (SPL)*. IEEE, 133–137.

[30] H. Reisinger, O. Blank, W. Heinrigs, A. Muhlhoff, W. Gustin, and C. Schlunder. 2006. Analysis of NBTI degradation- and recovery-behavior based on ultra fast VT-measurements. In *IEEE International Reliability Physics Symposium*. 448–453. DOI: http://dx.doi.org/10.1109/RELPHY.2006.251260

[31] P. Hehenberger, H. Reisinger, and T. Grasser. 2010. Recovery of negative and positive bias temperature stress in pMOSFETs. In *IEEE International Integrated Reliability Workshop Final Report*. 8–11. DOI: http://dx.doi.org/10.1109/IIRW.2010.5706473

[32] I. C. Nam, H. T. Kim, H. S. Park, K. S. Kim, K. H. Kim, J. B. Choi, J. U. Lee, S. W. Kim, G. C. Kang, D. J. Kim et al. 2004. Modeling of the substrate current and characterization of traps in MOSFETs under sub-bandgap photonic excitation. *J. Kor. Phys. Societ.* 45, 2 (2004), 1283–1287.

[33] N. D. Arora and M. S. Sharma. 1991. MOSFET substrate current model for circuit simulation. *IEEE Trans. Electron. Dev.* 38, 6 (1991), 1392–1398. DOI: http://dx.doi.org/10.1109/16.81631

[34] Haile Yu, Qiang Xu, and Philip H. W. Leong. 2010. Fine-grained characterization of process variation in FPGAs. In *International Conference on Field-programmable Technology*. 138–145. DOI: http://dx.doi.org/10.1109/FPT.2010.5681770

Received June 2021; revised October 2021; accepted February 2022