Physical Unclonable Functions Based on Delay Paths and an Interdigital Microstrip Notch Filter

Mitch Martin

University of New Mexico

Follow this and additional works at: https://digitalrepository.unm.edu/ece_etds

Part of the Electrical and Computer Engineering Commons

Recommended Citation


This Dissertation is brought to you for free and open access by the Engineering ETDs at UNM Digital Repository. It has been accepted for inclusion in Electrical and Computer Engineering ETDs by an authorized administrator of UNM Digital Repository. For more information, please contact disc@unm.edu.
Mitchell Martin
Candidate

Electrical and Computer Engineering
Department

This dissertation is approved, and it is acceptable in quality and form for publication:

Approved by the Dissertation Committee:

Jim Plusquellic, Chairperson

Payman Zarkesh-Ha

Jedidiah Crandall

Todd Bauer
Physical Unclonable Functions Based on Delay Paths and an Interdigital Microstrip Notch Filter

by

Mitchell T. Martin

B.S., Computer Engineering, University of New Mexico, 2009
M.S., Computer Engineering, University of New Mexico, 2011

DISSERTATION

Submitted in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy
Engineering

The University of New Mexico
Albuquerque, New Mexico

May, 2020
This dissertation is dedicated to my loving family, my father Steven ("Bubba") and Lynn, my loving mother Francine ("Bucky") and stepdad Doug ("Dougie") and my big brother Benjamin. The love and motivation that you all have given me over these past years is something that I am forever grateful for. Lastly, to my two amazing dogs Smokey and Bandit... Lets go for a walk.
Acknowledgments

As I sit here and think about these last 9 years of my education that have brought me to this point, I am filled with joy knowing the countless number of friends and colleagues I have gained.

I am forever grateful and always humbled by my advisor, Dr. Jim Plusquellic, whose knowledge and passion for his research continues to motivate me to become a stronger researcher. His dedication to his students, ability to systematically deconstruct problems, and efficiency are all traits I hope to one day master myself. Dr. Payman Zarkesh-Ha and Dr. Jedidiah Crandall of the University of New Mexico, I wholeheartedly appreciate the knowledge and skills I have gained from your teaching over the years. Dr. Todd Bauer of Sandia National Labs, I would also like to extend my sincere appreciation for the challenging work and knowledge I have received over the years, especially on those early projects I worked on as an intern.

In addition, my good friends from our past and present research team, Ian Wilcox, Dr. Jim Aarestad, Dr. Ryan Helinski, for their inspiration and excellence you all have shown me for research. My professional Colleagues Jason Hamlet, Mario Urrea, and countless others, thank you. Lastly, Christine Shields, thank you for believing in me and motivating me to do the best work that I can.
Physical Uncloneable Functions Based on Delay Paths and an Interdigital Microstrip Notch Filter

by

Mitchell T. Martin

B.S., Computer Engineering, University of New Mexico, 2009
M.S., Computer Engineering, University of New Mexico, 2011
PhD., Engineering, University of New Mexico, 2020

Abstract

A Physical Uncloneable Function (PUF) is an integrated circuit hardware primitive that is designed to leverage naturally occurring variations to produce a random bitstring. The Aribiter (ARB PUF) is one of the first to be described in the literature. It derives its entropy from variations that occur in the delays of identically configured logic paths. The ARB PUF uses a phase comparator to decide which path of a pair is faster under a given challenge, and generates a 0 or 1 as a response indicator bit. Unfortunately, the ARB PUF is not reliable, requiring error correction in cases where the sequence of response bits (the bitstring) needs to be reproduced. In this proposal, a test structure is described, called a time-to-digital converter (TDC) that is capable of measuring the actual delays of the paths. This type of ’soft’ information can be used to improve the reliability of the ARB PUF. Data obtained from a set of chips fabricated in IBM’s 90 nm technology, and collected across 9 temperature-voltage (TV) corners, is used to demonstrate its effectiveness.
Current PUF designs are typically implemented in silicon like the ARB PUF or utilize variations found in commercial off-the-shelf (COTS) parts. Because of this, existing designs are insufficient for the authentication of Printed Circuit Boards (PCBs). In this thesis, we also propose a novel PUF design that leverages board variations in a manufactured PCB to generate unique and stable identifiers (IDs) for each PCB. In particular, a single copper trace is used as a source of randomness for bitstring generation. The trace connects three notch filter structures in series, each of which is designed to reject specific but separate frequencies. The bitstrings generated with both the ARB PUF and the PCB PUF are evaluated using statistical tests which measure randomness, uniqueness and reliability.
Contents

List of Figures xii

List of Tables xv

1 Introduction 1

1.1 Overview 1

1.2 PUF Classification and Types 2

1.2.1 Strong PUFs 3

1.2.2 Weak PUFs 3

1.2.3 Delay-based PUFs 4

1.2.4 Ring-Oscillator PUFs 4

1.2.5 SRAM and Butterfly PUFs 4

1.2.6 Metal Resistance PUFs 5

1.3 Metrics for PUFs 5

1.3.1 Uniqueness 6
Contents

1.3.2 Reliability (Reproducibility) ........................................... 7

1.3.3 Randomness ................................................................. 7

1.4 Organization ................................................................. 8

2 Arbiter PUF with High Resolution Measurement Structure ........ 10

2.1 Introduction ........................................................................ 10

2.2 Challenges to PUF Design ................................................ 11

2.3 Chip Design ................................................................. 12

2.3.1 The ARB PUF .............................................................. 12

2.3.2 The TDC Structure ....................................................... 15

2.4 Experimental Results ..................................................... 19

2.4.1 TDC Sensitivity .......................................................... 19

2.4.2 Calibration ................................................................. 20

2.4.3 Usage for the ARB PUF and TDC ................................. 22

2.4.4 Applied Challenges .................................................... 23

2.4.5 Measurement and TV Noise Analysis ......................... 26

2.4.6 BitString Generation Technique .................................... 27

2.5 Analysis ................................................................. 29

2.5.1 Uniqueness ............................................................... 29

2.5.2 Reliability .............................................................. 30

2.5.3 Randomness ............................................................ 31
Contents

2.6 Conclusion ...................................................... 31

3 NotchPUF ......................................................... 32

3.1 Introduction .................................................. 32

3.2 Related Work ................................................ 34

3.3 Interdigital Notch Filter ...................................... 37

3.3.1 Interdigital Resonator .................................. 38

3.3.2 Double Interdigital Resonator ......................... 38

3.3.3 NotchPUF .................................................. 39

3.4 Simulations ..................................................... 41

3.4.1 Tuning ...................................................... 41

3.4.2 NotchPUF Simulations and Validation in Hardware 43

3.5 PCB Variation .................................................. 44

3.5.1 Board Thickness Variations ......................... 47

3.5.2 Trace Width Variation .................................. 48

3.5.3 Substrate Relative Dielectric Constant Variation 50

3.5.4 Summary .................................................. 53

3.6 Experimental Results ....................................... 54

3.6.1 Experimental Setup ..................................... 54

3.6.2 Signal Analysis and Bitstring Generation .......... 56

3.6.3 Bitstring Analysis ....................................... 60
Contents

5.4 Attack Threat Analysis and Mitigation .......... 86

5.5 Detailed Defect Analysis ........................ 86

6 Conclusions ........................................ 88
List of Figures

2.1 Basic Arbiter PUF . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
2.2 Layout of ARB PUF and TDC. . . . . . . . . . . . . . . . . . . . . 14
2.3 Arbiter PUF configuration showing Tap Points (TPs) . . . . . . . . 14
2.4 Time-To-Digital Converter with Path Select/Pulse Generator front-end for interfacing with the Arbiter PUF TP. . . . . . . . . . . . . . 16
2.5 Current Starved Inverter . . . . . . . . . . . . . . . . . . . . . . . 17
2.6 TDC Sensitivity . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20
2.7 TDC Cal1 vs. thermometer code curves across 9 TV corners on a chip. 21
2.8 Examples showing configurations for timing path segments of length 1 and 3 in the ARB PUF . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
2.9 TDC-timed path delay differences from 4 chips for paths of length 1 across Segment 7 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
2.10 Measurement noise analysis under different path selection criteria . . 26
2.11 Thresholding Technique . . . . . . . . . . . . . . . . . . . . . . . 28
2.12 Inter-Chip Hamming Distance Distribution for 32 Chips. . . . . . . 29
List of Figures

2.13 NIST Statistics ......................................................... 30

3.1 Interdigital Resonator with Equivalent circuit ..................... 37

3.2 PCB structure of a single notch filter with a double microstrip inter-
digital resonator ......................................................... 38

3.3 PCB structure of a single NotchPUF ................................ 40

3.4 Double Interdigital Resonator ........................................ 41

3.5 $S_{21}$ Notch filter tuning .............................................. 42

3.6 NotchPUF Simulation vs Measurement ................................ 44

3.7 NotchPUF Board Measurement Locations ........................... 47

3.8 Board Thickness Graph ................................................. 48

3.9 Trace Thickness Graph ................................................. 49

3.10 Two Microstrip Dielectric Constant Experiment ................... 50

3.11 Experimental Test Setup ............................................... 55

3.12 Frequency and Insertion Loss for NotchPUF PCBs ................ 57

3.13 2 GHz Modulus Operation Results .................................. 58

3.14 Inter-Distance .......................................................... 61

3.15 Intra-Distance .......................................................... 61

3.16 Changes in Er with Temperature ..................................... 62

4.1 All PCB Boards Considered ............................................ 64

4.2 Board Thickness Separated by Panel .................................. 66
List of Figures

4.3 Canny Filter Output .............................................. 67
4.4 Trace Width Measurement Region ................................ 68
4.5 Multi-panel Trace Width Variations ............................ 69
4.6 1GHz Multi-panel Structure Response .......................... 72
4.7 2GHz Multi-panel Structure Response .......................... 73
4.8 3GHz Multi-panel Structure Response .......................... 74
4.9 Modulus of Raw PUF Response .................................. 76
4.10 174 PCB Inter-Distance .......................................... 77
4.11 174 PCB Intra-Distance .......................................... 77
4.12 Combined Trace Defect .......................................... 79
4.13 Missing Trace Defect ............................................. 80
4.14 Copper Trace Pour Defect ...................................... 80
4.15 B34H1_2G vs B36H1_2G .......................................... 81
4.16 Copper Trace Factory Damage .................................. 82
4.17 Bad Panel ......................................................... 82
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Challenges Used</td>
<td>24</td>
</tr>
<tr>
<td>3.1</td>
<td>Simulation Properties</td>
<td>43</td>
</tr>
<tr>
<td>3.2</td>
<td>Average Measurement of $\Delta \phi$ (degrees) and $\varepsilon_{eff}$ Variations in FR-4 ($\varepsilon_r=4.8$) Substrate</td>
<td>52</td>
</tr>
<tr>
<td>3.3</td>
<td>Variation of $\varepsilon_r$ between boards in FR-4 ($\varepsilon_r=4.8$) Substrate</td>
<td>53</td>
</tr>
<tr>
<td>3.4</td>
<td>NotchPUF Parameter Results</td>
<td>57</td>
</tr>
<tr>
<td>4.1</td>
<td>PCB Panel Board Thickness Results</td>
<td>66</td>
</tr>
<tr>
<td>4.2</td>
<td>PCB Panel Trace Width Results</td>
<td>70</td>
</tr>
<tr>
<td>4.3</td>
<td>Multi-board Variation of $\varepsilon_r$ in FR-4 ($\varepsilon_r=4.8$) Substrate</td>
<td>70</td>
</tr>
<tr>
<td>4.4</td>
<td>Multi-panel 1GHz Structure Response Average</td>
<td>72</td>
</tr>
<tr>
<td>4.5</td>
<td>Multi-panel 2GHz Structure Response Average</td>
<td>73</td>
</tr>
<tr>
<td>4.6</td>
<td>Multi-panel 3GHz Structure Response Average</td>
<td>74</td>
</tr>
<tr>
<td>4.7</td>
<td>Multi-panel Mean Inter and Intra HD</td>
<td>76</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Overview

The idea of using intrinsic random physical features for identification is a promising solution for hardware security. Fingerprint identification of humans dates at least back to the nineteenth century \[23\] and led to the field of biometrics. Later on in the eighties and nineties, random patterns in paper \[5\] and optical tokens \[66\] were used for unique identification. These concepts were later formalized in the very beginning of the twenty first century, first as physical one-way functions \[50\], physical random functions \[13\] and finally as PUFs. In the years following, an increasing number of new types of PUFs were proposed, with a tendency towards more integrated constructions. The practical relevance of PUFs for security applications was recognized from the start, with a special focus on the properties of physical unclonability, entropy extraction and tamper resistance.

PUFs are poised to represent the next generation of hardware security primitives for integrated circuits. The chip-specific identifiers produced by PUFs can serve several applications including chip ID, authentication, metering and encryption. PUFs:
Chapter 1. Introduction

measure and digitize the natural variations that occur in path delays, leakage current, Static Random Access Memory (SRAM) power-up patterns, etc., to produce a long sequence of random bits, i.e., a bitstring. Most of the applications that use these bitstrings require that they be 1) unique among the chip population, 2) random in sequence and 3) reproducible across environmental conditions.

Over the last few years, interest in PUFs has increased substantially including the number of various implementation types. Due to the robustness of PUFs, the security community has also taken notice and are commonly used in this field. The advantage a PUF has over traditional key storage methods like eFuses, electrically erasable programmable read-only memories (EEPROMs), Battery Backed Ram, etc., is that PUFs take advantage of inherently unclonable physical structures (due to process variations). This therefore makes a PUF easy to evaluate, but hard to predict.

1.2 PUF Classification and Types

Due to the amount of attention PUFs have received in recent years, many different types of PUF primitives have been proposed. The diversity in PUFs is also what gives it the ability to be adopted for many different types of applications. Whether you want to utilize pre-existing circuitry, or something completely custom, chances are exists a primitive that can be used. Most proposed PUFs fall into one of several categories: delay chains and ring-oscillator (RO) PUFs [12] [41] [52] [39], SRAM PUFs [13] [60], metal resistance PUFs [21] [31], Butterfly PUFs [32], and many others.

The classification type for a given PUF stems from the security properties of their challenge-response pairs. The following sections 1.2.1 and 1.2.2 describe this in more detail. Further, the aforementioned types of PUFs are discussed in more detail in...
1.2.1 Strong PUFs

A strong PUF should support a large number of Challenge-Response Pairs (CRPs) and a complete measurement of all CRPs within a decent amount of time. Further, it should be difficult for an adversary to predict the response even with prior knowledge of circuit design and a number of CRPs. This also implies that the PUF be resilient against model-building type attacks. Thus making it difficult to mimic/clone the behavior of a strong PUF. These features are critical for applications such as Integrated Circuit (IC) identification and secret key generation. Some examples of a strong silicon PUF are Arbiter PUFs, feed-forward Arbiter PUFs [35], XOR Arbiter PUFs [63], and lightweight secure PUFs [41].

1.2.2 Weak PUFs

Weak PUFs on the other hand support a limited number of CRPs (even a single challenge at times). This makes them unsuited for use in IC authentication and also due to the possibility of a replay attack. Weak PUFs are often used for secret key generation in non-critical cryptographic systems. Rather than storing secret keys in non-volatile memory, weak PUFs offer an easy means in which to generate keys. Because of this, invasive key recovery techniques become more difficult [54] rather than being able to just dump the contents of non-volatile memory. This doesn’t solve all problems, as they are still susceptible to side channel attacks [37]. Typical examples of weak PUFs are SRAM PUFs, and the butterfly PUFs.
1.2.3 Delay-based PUFs

A delay-based PUF makes use of variations that are introduced at time of manufacturing. These types of circuits involve extraction and comparison of delay differences of signals. These delay differences are utilized in multiple different ways in which entropy extraction can be maximized. The most common and first to appear of these circuits is the RO PUF.

1.2.4 Ring-Oscillator PUFs

The RO PUF is based on a delay loop, typically a number of inverters strung together with the end feeding back, thus creating the loop. The benefit of an RO PUF because of their sensitivity to process variations, is that they have been widely used in modeling process variations with good results. They are also not very efficient in generating a decent amount of challenge response pairs. With one pair of ROs, a single bit can be extracted by a simple comparison of which RO had a higher (or lower) frequency. Because of this, many copies are created so that approx $\log_2(n!)$ comparisons can be made. Although due to the complexity and design challenges of most PUFs, the RO is very attractive due to it’s simplicity. Unfortunately, because of the simplistic design of the RO some challenges arise such as model-building attacks, dynamic power consumption, and area costs (due to multiple copies).

1.2.5 SRAM and Butterfly PUFs

SRAM PUFs are based on random start-up states of a traditional 6 transistor static RAM cell, which was first proposed as a system of fingerprint extraction and random numbers in SRAM. This was later implemented by introducing the idea of intrinsic PUFs for Field Programmable Gate Arrays (FPGAs) Intellectual Property
Chapter 1. Introduction

(IP) protection [16]. This was later extended for [FPGA] automatically initialize internal [SRAM] to all zeros at startup. [32].

The Butterfly [PUF] is instead based upon two opposing NAND gates that interact in a similar way as an [SRAM] cell where the variation is in the threshold voltage of transistors. A Butterfly [PUF] cell employs two cross-coupled latches, and exploits the random assignment of a stable state from an unstable state that is forcefully imposed by holding one latch in preset while the other in clear mode by an excite signal.

1.2.6 Metal Resistance PUFs

The metal resistance-based [PUF]s derive their entropy from random physical variations in the metal contacts, vias and wires that define the power grid and interconnects of an [IC] [21] [31] [30]. Two critical VLSI wiring wear-out failure mechanisms are electromigration (EM) [58] and mechanical stress-induced voiding (SV) [26]. However, these effects are well understood and can be completely avoided with proper sizing of the wires, vias and contacts.

1.3 Metrics for PUFs

For any [PUF] to be considered “reliable” they must satisfy a suite of various statistical performance metrics. The authors of [29] proposed a set of performance metrics that are based on delay statistics. The 3 main metrics that are used are Uniqueness, Reliability (which I refer to as “Reproducibility”), and Randomness. Uniqueness indicates the entropy between two [PUF]s, either in the same device (intra-device) or between devices (inter-device). The Reproducibility expresses the level of [PUF] reliability which is decreased by the noise coming from the measurement environment.
Chapter 1. Introduction

and temperature/voltage variations. Randomness indicates if there is an imbalance between the number of '0’s and '1’s in the bitstrings for all of the challenges. A well-defined National Institute of Standards and Technology (NIST) statistical test suite [55] is used to perform randomness tests on generated bitstrings in this work.

1.3.1 Uniqueness

In order for a PUF to be unique, it should be different among others in the same population. The primary metric used is “inter-device” variation. Inter-device variation is derived using pair-wise Hamming distance comparisons between two sequences of PUF output bits from different PUF devices. The average inter Hamming Distance (HD) for a group of m PUF instances can be computed using the following formula:

\[
\text{Inter} \text{HD}_{\text{avg}} = \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{HD(R_w,R_v)}{n} \times 100\%
\]

(1.1)

\(HD(R_w,R_v)\) represents the hamming distance between the responses generated by \(u^{th}\) and \(v^{th}\) PUF instances for a given challenge. An ideal Hamming distance distribution would indicate a mean tightly grouped around 0.5. This would indicate uniqueness between every generated PUF response in the population. The identification capability of a PUF is directly related to the amount of process variation, specifically intra-chip variation present. Large within-chip process variation results in a larger value of uniqueness.
1.3.2 Reliability (Reproducibility)

A robust PUF circuit is one that should be capable of reliably reproducing Challenge-Response Pair (CRP) in the presence of noise and environmental variations. Supply voltage variations and temperature variations adversely affect the delay and power consumption of a circuit and it may affect various parts of the circuit differently. Similar to uniqueness, reliability can also be measured by calculating the average Intra-chip Hamming Distance (Intra-HD). For a particular PUF instance, intra-HD refers to the Hamming Distance between the two measured responses when the same challenge is applied twice. The average intra-HD for a group of m PUF instances can be computed using the following formula 1.2 [38]:

\[
\text{Intra}_{HD_{avg}} = \frac{1}{x} \sum_{y=1}^{x} \frac{HD(R_i, R'_{i,y})}{n} \times 100\% \quad (1.2)
\]

Where \(HD(R, R')\) is over x samples, for the PUF instance i. \(R'_{i,y}\) is the \(y^{th}\) sample of \(R_i\). The ideal value of the intra-HD is 0%, which implies that the original PUF responses can be reproduced with zero error bits across all the sampled conditions. Any error produced can result in different responses for the same challenge from a given PUF instance. Most PUF circuits use relative comparison to generate CRPs achieving a high degree of reliability. This is measured by looking at the total number of bit errors in responses obtained by subjecting the PUF to different voltage and temperature conditions. An ideal PUF should be able to reliably reproduce a response at all environmental conditions without the need for error correction.

1.3.3 Randomness

For a bitstring of length \(n\) to be random, the probability of predicting bit \(n+1\) is 50%. A suite of statistical tests also exists to help determine the level of randomness.
Chapter 1. Introduction

of a bitstring \[55\]. A high level of randomness is important because true random number generators (TRNG) are essential for use in cryptographic applications \[57\]. A quick sanity check test that is performed for this metric is bitstring bias. That is, a comparison of the number of '1’s to '0’s in the bitstring. Ideally this should be as close as possible to 50%. This may be good for a simple test but is not enough to confidently describe a bitstring as being truly random. A much more in depth approach to this determination is to use the well-developed suite of tools from NIST. These tools look at a wide range of different bitstring features such as, frequency (monobit) test, longest run of ones in a block, non-overlapping template matching test, approximate entropy test and many others. Only a subset of these tests were performed in this research and is explained in more detail in the following sections.

1.4 Organization

This dissertation is organized as follows:

In this Chapter 1, a PUF introduction and overview was presented. The readers were introduced to various types of PUFs that currently exist along with their associated strengths and weaknesses. The standard PUF metrics were also described. These metrics are a measure for the uniqueness, reliability and finally the randomness for the generated PUF bitstrings.

Chapter 2 presents an Arbiter PUF with a high resolution measurement structure. The manufactured 90 nm IBM chip design is presented by describing both the ARB PUF and TDC structure. The experimental results follow, with a discussion on the TDC sensitivity and the calibration method used for the structure. We then describe the results of the measurement and noise analysis and finally the bitstring generation technique we used. We analyzed the results from multiple temperature voltage corners to ensure the aforementioned PUF metrics could be satisfied.
Chapter 1. Introduction

Chapter 3 moves on to a new PUF structure for printed circuit boards. First the introduction into the NotchPUF, followed by a discussion about current and relevant work that exists. The interdigital notch filter structure is then introduced with a discussion detailing the characteristics of the resonator. Simulation results of tuning an individual resonator and the combined NotchPUF structure are then described. Following this, an investigation related to the level of variations that exist on the manufactured PCBs is described. Finally, the experimental results of the NotchPUF are shown and concludes the this chapter.

In Chapter 4, multiple panels of NotchPUF PCBs are produced and analyzed. First by taking a look at the how the variations change on separate panels of PCBs from the same manufacturer. The NotchPUF experiments and analysis are then repeated using a total of 174 PCBs. The PUF metrics are used once again to evaluate the behavior of the NotchPUF. Finally, a closer look is taken at the various PCBs that were manufactured and how it affects their response.

In Chapter 5, I discuss future work and in chapter 6, I summarize the findings and contributions of this dissertation.
Chapter 2

Arbiter PUF with High Resolution Measurement Structure

2.1 Introduction

In this chapter, we propose a new Arbiter (ARB) PUF primitive and a high speed time-to-digital converter (TDC) structure. This Arbiter PUF is based upon the delay of identically configured paths. A delay PUF exploits the random variations in delays of wires and gates on silicon [65]. Given an input challenge, a race condition is set up in the circuit, and two transitions that propagate along different paths are compared to see which comes first. An arbiter, typically implemented as a latch, produces a 1 or a 0, depending on which transition comes first. Due to process variations during manufacturing, there exists differences in propagation delays of each path along the identically laid out Multiplexer (MUX) chain. Traditional Arbiter PUFs only compare the path delay difference at the end with an Arbiter element [45]. In this scheme, only 1 bit of entropy is extracted based upon which signal propagated the quickest. In the proposed primitive, a set of “tap-points” has been
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

included throughout the delay chain which increases the amount of entropy that can be extracted. In addition to these set of tap points, the TDC provides a method in which precise delay information can be generated. This delay information produces “soft” data which enhances the ability to perform reliable regeneration.

A set of statistical tools is used to evaluate the PUF’s randomness and reliability. The majority of these tests come from the National Institute of Standards and Technology (NIST) [56]. In addition to these tools are a set of metrics that are used to determine a PUF’s predictability, reliability and susceptibility to reverse engineering [42].

2.2 Challenges to PUF Design

The core challenge for all PUFs is to extract features that are derived from naturally occurring conditions and to make them meaningful. Each of the PUFs currently proposed and implemented all share some design challenges in one way or another. Whether it is from measurement noise, environmental changes, wear out or even aging effects, additional functionality must be considered to minimize these effects. The implementation of a technique or method to solve these issues, could be considered almost as important as the PUFs themselves.

The goal that all PUFs strive to achieve is that of perfect randomness and error tolerance with no overhead. With current implementations and proposals, overhead can be described as the use of some sort of “helper data” that is used to correct error prone bits. There are different forms of helper data, but the overall usage for it remains the same. Having helper data leads to another challenge of minimizing the amount of data that you are revealing to the public. Helper data is inherently made public to be used such that a regenerated bitstring can be error corrected. Helper can also be used by an adversary, so by decreasing that amount of helper
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

data present, the less that is revealed about the design. A PUF in the ideal world would have no need for helper data, thus minimizing the amount of leakage.

One technique that helps with this problem is by using some form of redundancy, upon which a voting scheme is also implemented. This voting scheme would then provide a means in which the probability of a single-bit error in a bitstring is reduced. Unfortunately this also incurs a penalty in terms of area and power consumption, while maximizing the use of available entropy in the PUF design.

Previous solutions for the delay-based ARB PUF have been to extract only a single bit for the delay path that had “won” the race against the identical opposite path. Our novel solution for the ARB PUF offers the following characteristics: it has multiple tap points which are used to extract the maximum amount of entropy compared to other basic implementations, path delays are digitized such that more error prone bits are more easily apparent, because of the digitization, a simple thresholding technique can be used to recover an error-free bitstring, and all components of the PUF (minus layout) can be implemented with standard library components such as inverters and 2-to-1 MUXes.

2.3 Chip Design

2.3.1 The ARB PUF

The architecture of the proposed PUF consists of two basic components; an ARB PUF, which implements the paths to be tested, and a TDC, which provides high resolution timing measurements of the path delays in the ARB PUF. The following presents the implementation details of the ARB PUF component.

A typical ARB PUF consists only of two delay paths that have an identical layout
path and length [63]. This delay path includes a matching set of Multiplexers that are used to pass through the two delay signals from the left side of the input. The path chosen is determined by the selection of challenge bits which control the MUXes. After the signals race through the two delay paths, the arbiter (latch) at the end decides which signal is faster. The output is a '1' if the signal to the latch input is faster, and '0' otherwise. The output of the Arbiter is called the response as shown in Fig 2.1. This single response multiple challenge type PUF is very limited and lacks in its ability to fully extract entropy.

Figure 2.1: Basic Arbiter PUF

The proposed Arbiter PUF design utilizes a set multiple tap points along the delay chain, which extracts a greater amount of entropy. Fig. 2.2 shows the layout of the ARB PUF and TDC in the 90 nm test chip architecture. The ARB is shown along the top as a sequence of 16 series-connected segments of 8 elements each. These tap points are on both the upper and lower MUXes in the delay chain. With this architecture, the pulse is able to be inspected at essentially the same distance of propagation. This also allows comparison of alternate paths of the same length.

Fig. 2.3 gives a schematic level representation of the elements within the ARB. Each of the 128 elements consist of a flip-flop (FF) and two copies of a 2-to-1 MUX.
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

The FF is scan-connected with the others (not shown) and can be configured with a challenge bit. The challenge bit determines whether two paths (labeled $P_A$ and $P_B$) propagate signals straight through the 2-to-1 MUXes (when 0) or cross-over with $P_A$ propagating through the bottom MUX and $P_B$ through the top (when 1). The input to the ARB PUF is shown on the left side of Fig. 2.3 and connects to both of the $P_A$ and $P_B$ paths. A signal transition is introduced into the ARB PUF by asserting or de-asserting this input signal.

A unique feature of the ARB PUF proposed in this work as described earlier, is the introduction of a set of 'tap points' (several are labeled in Fig. 2.2). The first tap point is connected directly to the input of the ARB. The remaining tap points are implemented by fanning out at specific points along the paths $P_A$ and $P_B$ to a pair of buffers. For example, the second tap point connects to $P_A$ and $P_B$ at a point.
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

that is 32 elements from the input side of the Arbiter PUF. The remaining 6 tap points connect at points further downstream after sequences of 16 additional elements. The outputs of the buffers at each tap point route to the inputs of the TDC.

2.3.2 The TDC Structure

The TDC is designed to measure the relative delay between two input signals which are provided by a pair of tap points on the Arbiter PUF. The relative delay is digitized by the TDC using a pulse-shrinking mechanism (described below). The digital code is ’scanned out’ of the TDC for post-processing.

The TDC is implemented as two components, labeled Path Select/Pulse Gen Unit and Pulse Shrinking Delay Chain in the layout of Fig. 2.2 and in the schematic of Fig. 2.4. Scan FFs in the Path Select/Pulse Gen Unit, labeled ’Sel A’ and ’Sel B’, drive the inputs of two 8-to-1 MUXes, which, in turn, select a specific pairing of tap point inputs, one from group ’A’ and one from group ’B’. The outputs of the 8-to-1 MUXes route to the inputs of an XNOR gate, which serves to generate a negative pulse for the Pulse Shrinking Delay Chain on the right (see annotation in Fig. 2.4). The arrival of an edge on one of the tap points propagates to the XNOR and generates the 1-to-0 transition of this negative pulse, and an edge (arriving later) on the second tap point generates the 0-to-1 transition of the pulse. Specific configurations of tap point pairs that provide sufficient skew between the two edges from paths in the Arbiter PUF are described in the Applied Challenges section.

The TDC is designed to ’pulse shrink’ the negative output pulse from the XNOR as it propagates down a current-starved inverter chain. This pulse shrinking delay chain is a modified version of the one used by [3]. Two cells are depicted in Fig. 2.5. During the time when the input of the circuit in Fig. 2.5 is high, the current available to discharge the parasitic capacitance C1 seen from point B to ground is
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

Figure 2.4: Time-To-Digital Converter with Path Select/Pulse Generator front-end for interfacing with the Arbiter PUF TP.

the current through M5. This current determines the delay between turning on M2 and turning off M4. It can be observed that this variable delay acts on the leading edge of the waveform at the output of the inverter M2-M4. Similarly applies to M6 with respect to the rate of discharge of the parasitic capacitance C2, with the difference that now the current through M6 controls the delay of the trailing edge of the waveform at the output of the circuit. If this cell is part of a chain of n similar cells (as is the case in this proposal), a pulse delayed by \( nt \) with respect to the input appears at the output, where \( t \) is the delay of the cell represented in Fig. 2.5. This delay is controlled by the gate voltages of M5 and M6. The range of the delay can be controlled by appropriately sizing the transistors in the circuit. This circuit is used in the proposed TDC to appropriately propagate and shrink a pulse from the input.

As a pulse moves down the inverter chain in the TDC, it activates a corresponding set of set-reset latches to record the passage of the pulse, where activation is defined as storing a '1'. A thermometer code, i.e., a sequence of '1's followed by a sequence of '0's, represents the digitized delay of a path within the Arb PUf. We refer to the number of '1's in the Thermometer Code Voltages (TCV) in the following sections.

A single cell of the current-starved inverter used in the delay chain is shown on the
far right side of Fig. 2.4 and two in series in Fig. 2.5. The NFET transistor with input labeled 'Calx' implements the current-starving mechanism. The Calx inputs are driven by two analog control voltages, labeled 'Cal0' and 'Cal1'. The current-starved inputs of all the even numbered inverters (numbered starting with 0) are connected to Cal0 while the inputs of the odd numbered inverters are connected to Cal1. This type of configuration allows independent control over the propagation speed of the two transitions associated with the negative pulse. For example, increasing the voltage on Cal1 toward the supply voltage allows the odd numbered inverters to switch more quickly when the first transition, i.e., the 1-to-0 input transition, propagates to their inputs. Note that the 1-to-0 input transition creates 0-to-1 transitions on the inputs of the odd numbered inverters in the chain, which activates the pull-down paths of these inverters. With Cal0 fixed at a specific voltage, larger assigned Cal1 voltages allows the pulse to 'survive' longer in the delay chain because the first edge propagates more quickly. The speed of trailing 0-to-1 input transition does not change with Cal0 fixed, and therefore it takes longer for this edge to catch-up to the
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

leading transition. Eventually it does, (assuming Cal0 and Cal1 are set such that the trailing edge is faster), and the pulse disappears. All latches up to the point where the pulse disappears store a ’1’, while those beyond this point store ’0’. The state of the latches can then be transferred to the scan FFs shown along the bottom of Fig. 2.4 for scan-out and analysis.

The pulse-shrinking behavior of the TDC allows very high timing resolution, i.e., 10’s of picoseconds, in measurements of the width of the input pulse assuming the Cal0 and Cal1 voltages are fixed and stable. The timing resolution of the TDC is related to how far the pulse propagates along the delay chain, where pulse propagations to points near the end of the delay chain provide the highest resolutions. It is possible, however, for the pulse to propagate off the end of the TDC, a condition referred to as overflow, which obviously must be avoided. By choosing the proper Calx voltages, the overflow condition can be prevented while simultaneously allowing for high timing resolutions.

In our experiments, both of these voltages are controlled using off-chip power supplies. This allowed us to explore the parameters of this new architecture so that a functional and fully integrated version can be implemented properly on the next test chip. The off-chip power supplies will be replaced with an on-chip Digital-to-Analog Converter (DAC), and a controller will be used to select the proper Calx voltages from this DAC. This future circuit will also be resilient to environmental changes so a steady Calx voltage can be produced. As discussed in the following sections, the primary function of the controller will be to carry out a calibration process that is designed to prevent overflow. From our experiments, maximizing the timing resolution is of benefit but is not a requirement for the TDC to be effective in improving reliability of the Arbiter PUF.

In preliminary experiments, we discovered that it is not necessary to have independent control over the leading and trailing edges of the pulse. The data presented
here is obtained by fixing Cal0 to the supply voltage. Therefore, only Cal1 is tuned in our experiments. The Cal1 voltage required to meet the above constraints varied as a function of the ambient temperature and voltage conditions but was largely self-compensating. We provide more details on this issue after we describe how the Arbiter PUF and TDC are used together to collect delay measurements in the next section.

The overhead of the proposed Arbiter + TDC combination is as follows. The Arbiter PUF with 128 elements occupies an area of approx. 525 um x 25 um (13k um$^2$) while the TDC occupies an area of 176 um x 60 um (10k um$^2$). As we show in the following sections, the size of the Arbiter PUF is sufficient to generate several hundred delays, each of which has at least one constituent element in a given Arbiter delay path that is completely independent of the others. Simple modifications can be made to increase the number of independent delays to a 1000 or more with only a moderate increase in area.

### 2.4 Experimental Results

#### 2.4.1 TDC Sensitivity

Calx as shown on the right side of Fig. 2.4 is used to tune the resolution of the TDC. When tuning, there is a trade off between range and resolution. This trade off is shown in Fig. 2.6. This test was performed with a Xilinx ZYNQ FPGA which used the digital clock manager (DCM) to drive a pulse into the input of the TDC in Fig. 2.4. This DCM per Xilinx’s specs can be tuned to a resolution of about 18ps which is sufficient for this analysis. A delay from about 2ns to approximately 14ns is then measured at various Calx voltages. These pulses produce a TCV value which is shown along the y-axis from a range of 0 to 120. 6 different curves at 50mV
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

Figure 2.6: TDC Sensitivity

increments are illustrated. At the lower Calx voltage (600mV), a timing resolution of approx. 130ps per TC is produced and for higher values at 850mV, this resolution increases to approx. 20ps per TC. From this analysis, the TDC is able to achieve a timing resolution of about 25ps. This will allow the high resolution measurements that are needed for this delay-based PUF once calibration is completed.

2.4.2 Calibration

As indicated earlier, Cal1 needs to be ‘tuned’ to compensate for changes in the TDC behavior introduced by TV variations. The curves in Fig. 2.7 illustrate the behavior of the TDC in one of the chips at the nine TV corners investigated in this work. The x-axis sweeps the Cal1 voltage over a range of 450 to 750 mV. The y-axis plots the number of 1’s read from the TDC under each of these Cal1 settings. The individual curves are labeled to indicate the TV corner under which the data was collected. The
mean values as well as the $3\sigma$ upper and lower limits are superimposed. Although the shapes of the curves change to some degree, the main effect of TV variations is reflected as the shift in the curves along the x-axis.

In order to ensure that the [TDC] is able to produce values in the region labeled 'target region' (90 to 105 1’s from [TDC]) at each of these TV corners, it is necessary to 'tune' the Cal1 voltage. Note that shifts due to voltage variations will be automatically calibrated for by an on-chip DAC. This is true because the DAC will be connected to the power grid on the chip and will track changes in the power supply voltage automatically. Therefore, the primary issue is dealing with shifts introduced by temperature variations.

A calibration procedure is proposed to tune Cal1 so that overflow does not occur and the [TDC] produces values in the target region under temperature variations. The objective of the calibration process is to select a voltage produced by the on-chip DAC and apply this voltage to the Cal1 signal of the [TDC]. This can be accomplished by
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

Figure 2.8: Examples showing configurations for timing path segments of length 1 and 3 in the ARB PUF.

choosing a tap point combination and interactively testing that path and adjusting the voltage until the number of 1’s produced is in the target region. The process can be implemented by an on-chip state machine and using a binary search process (to make it fast). In our experiments, we emulate the binary search process in LABVIEW software and use an external power supply to emulate the on-chip DAC.

2.4.3 Usage for the ARB PUF and TDC

As covered above, the addition of the tap points provides a unique opportunity to measure delays along segments of the ARB PUF (traditional approaches do not allow entropy to be extracted from the constituent elements of the ARB's delay chains). The diagram in Fig. 2.8 illustrates how the tap points can be used to measure delays along path segments. The elongated rectangles represent an abstraction of the ARB PUF in which the 128 elements are partitioned into seven segments labeled 1 to 7. The first segment contains 32 elements while the remaining segments contain 16 elements. The top portion shows two configurations for measuring paths of length 1 (in segments 2 and 7) and one configuration for measuring paths of length 3 (across
Chapter 2. *Arbiter PUF with High Resolution Measurement Structure*

Segments 2, 3 and 4\(^1\). In the examples shown, the number of switches configured with a ‘1’ is odd, which ensures that the TDC times a single path. For example, the signal propagating along the top path to the tap point at the beginning of segment 2 crosses-over to the bottom path before reaching the second tap point at the beginning of segment 3. The path that is timed is highlighted in the figure. We use the term x-over to refer to switches that are configured to cause the path to cross-over from top to bottom or bottom to top.

In order to eliminate any bias that exists in the TDC measurement structure, in particular along the paths from the tap points through the 8-to-1 MUXes in the TDC and to the XNOR gate, complementary paths of those shown in the top portion of Fig. 2.8 are also tested and the two measurements are subtracted. The bottom portion of Fig. 2.8 shows the complementary configuration of the three tap point combinations given in the top portion.

As indicated above in reference to Fig. 2.8, other tap point configurations allow the measurement of delays from paths that traverse multiple segments. However, the statistical averaging effect of delays along longer path segments makes it difficult to measure distinguishing characteristics in them at sufficient resolution, and therefore, their usefulness for PUF bit generation is limited. Therefore, only paths of length 1 are used to generate the bitstrings analyzed in this proposal.

### 2.4.4 Applied Challenges

We applied a total of 40 distinct challenges to each chip and collected 211 data points from all paths of length 1. For the first 16 challenges, only a single ‘1’ is placed into each of the 6 segments labeled Seg2 through Seg7 from Fig. 2.8. The single ‘1’ creates one crossover (1-over) and allows paths of length 1 to be timed.

\(^1\) A path of length 1 is defined as a 16-element segment within the ARB PUF.
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

### Challenges Used

| 1) 6 (differences) per challenge for challenges 2-17 (total of 16*6 = 96) |
| 2) 6 (differences) per challenge for challenges 18-33 (16*6 = 96) |
| 3) 1 in Challenge 34 (TP 0 combo) (1) |
| 4) 4 in Challenge 35 (TP 0, 1, 4 and 5 combos) (4) |
| 5) 5 in Challenge 36 (TP 0, 1, 2, 4 and 5 combos) (5) |
| 6) 3 in Challenge 37 (TP 1, 3 and 5 combos) (3) |
| 7) 3 in Challenge 38 (TP 3, 4 and 5 combos) (3) |
| 8) 3 in Challenge 39 (TP 1, 2 and 5 combos) (3) |

Total: \( 96 + 96 + 1 + 4 + 5 + 3 + 3 = 211 \)

Table 2.1: Challenges Used

between the corresponding tap points. With 6 segments and 16 challenges, a total of 96 Thermometer Code Voltage Differences (TCVDs) can be generated. The next 14 challenges introduce 3 ‘1’s in each segment, producing 84 TCVDs. Challenges 31 and 32 introduce 7 and 15 ‘1’s respectively, yielding 12 TCVDs. The 8 remaining challenges were selected randomly and after analysis were found to time an additional 19 paths of length 1. The total number of TCVDs computed for each chip is 211. This is summarized in Table 2.1. These particular Tap Points (TP) combinations were chosen after data collection primarily because they provided a higher amount of entropy.

The waveforms shown in Fig. 2.9 depict TDC measurement results for paths of length 1 in Segment 7 for four chips. The waveforms are offset along the y-axis to facilitate comparisons between the waveforms. The path delays plotted along the y-axis are given in units of TCV bits. As indicated above, the values plotted are actually the difference in the number of TCV bits measured from two complementary paths. The plotted differences are computed using the average number TCV bits from a set of 11 measurements carried out on each path and its complement. The curves for each of the nine TV corners are superimposed to illustrate the ‘noise’ introduced by environmental variations. From the graphs, it is clear that TV variations are
The first 16 data points show the results from a set of canonical challenges. The canonical challenges introduce exactly 1 x-over, similar to those shown in Fig. 2.8 for paths of length 1. The data points are ordered so that the position of the x-over element in each test is adjacent to x-over elements that were tested under previous (and subsequent) challenges. This arrangement allows the magnitude of delay variations introduced by swapping a single pair of elements to be observed incrementally along each of the waveform segments. The next set of data points are arranged similarly except the consecutive tests introduce 3 x-overs. Although delay variations within these groups are relatively small, variations across groups and especially across segments are much larger. Also, even though only the delay differences for Segment 7 are shown, each previous segment produced similar results. In the experimental results section, we show that good statistical results can be obtained from these TCV measured delays.
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

2.4.5 Measurement and TV Noise Analysis

Measurement noise and noise introduced by varying temperature and voltage conditions work to reduce the reliability of the Arbiter PUF. Reliability is defined here as the ability of the Arbiter PUF to exactly reproduce the same bitstring during ‘regeneration’ experiments. The bitstrings produced at 25°C and at 1.20 V (nominal supply voltage) are referred to as the reference (or enrollment) bitstrings\(^2\) while bitstrings produced at the remaining 8 TV corners are referred to as regeneration bitstrings.

As indicated in the Introduction, the chips used in our experiments are tested at all combinations of temperatures −40°C, 25°C and 85°C and voltages 1.08 V, 1.20 V and 1.32 V. In this section, we evaluate these noise levels independently.

\(^2\)Enrollment defines the bitstring generation process that is carried out initially
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

The plots in Fig. 2.10 depict noise levels as 'average 3σ values' on the y-axis.

The plots labeled Fig. 2.10(a) and 2.10(b) depict 32 different waveforms, one for each of the thirty two chips considered in this analysis. The waveforms in Fig. 2.10(a) give the average 3σ’s of all measurements and challenges for each path length.

Fig. 2.10(b) shows variations introduced by changes in temperature and voltage. The y-axis in this case also plots the average 3σ’s of all path measurements at each of the nine TV corners, as indicated by the labels in the figure. Each point in Fig. 2.10(b) depicts the combined average of all 40 challenges and paths of length 1 for each chip. In general, noise levels are larger at −40°C (left portion of plot) than at 25°C and 85°C. Also, noise increases as supply voltage is lowered, as shown by the y-magnitudes of the points within each temperature group. For most chips, the noise contribution to the 3σ variation increases as the length of the path increases, which is also shown in Fig. 2.10(a) above. This is expected given that longer signal paths have larger amounts of jitter and are exposed over longer periods to power supply noise variations.

2.4.6 BitString Generation Technique

We applied the 40 challenges described in the Applied Challenges section to each of the chips and computed Thermometer Code Voltage Differences (TCVDs) from tests of complementary path pairings. The bitstrings are generated by comparing the 211 TCVDs obtained from each chip in all combinations, which yields bitstrings of length 22,155 bits. As noted below, a thresholding technique is used to discard those comparisons which are vulnerable to producing 'bit flips' under TV variations.

3Three σ is a statistical measure that bounds 99.73% of the population
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

Figure 2.11: Thresholding Technique

Thresholding Technique

The 'soft' information provided by the TDC can be used to avoid those path delay pairings whose difference is likely to result in a bit flip during regeneration. A thresholding technique is proposed that accomplishes this goal. During enrollment, comparisons of delay differences which are smaller than the threshold are discarded. The comparisons that are discarded are recorded in public data so that they are avoided during the regeneration process. Based on our analysis, we found that a threshold of approx. 5 (in units of TCVs and after absolute value) eliminates all bit flips that occur in the bitstring generation of our chips. Fig 2.11 illustrates the region in which sets of comparisons are discarded. The comparisons that fall outside of the threshold region have shown to be a sufficient distance from origin such that
upon regeneration, no bit errors are produced. 2.11 is from a sample dataset of comparisons from chip 2 at nominal conditions for illustrative purposes.

2.5 Analysis

In this section, we evaluate the several important statistical properties of the bit-strings including uniqueness and probability of bit flips, e.g., failures to regenerate the bitstring under different environmental conditions, reliability, and randomness.

2.5.1 Uniqueness

![Inter-Chip Hamming Distance Distribution for 32 Chips](image)

Figure 2.12: Inter-Chip Hamming Distance Distribution for 32 Chips.

The size of the bitstring after thresholding is 1,913 bits on average, which is approx. 11.5% of the 22,155 pairing combinations. The inter-chip HD requires that
Chapter 2. Arbiter PUF with High Resolution Measurement Structure

the bitstrings for all chips are the same size. So, we use the chip with the shortest bitstring and reduce the size of the other bitstrings to this length. The smallest bitstring is 1,282 bits. The HDs from the bitstrings of the 32 chips are computed under all combinations. Fig. 2.12 gives the inter-chip Hamming Distance (HD) distribution along with superimposed Gaussian curve fit to illustrate the level of conformance of the distribution to a normal distribution. The average inter-chip HD is given as 50.004%, which is very close to the ideal of 50%.

Only nine of the NIST tests are applicable to bitstrings of size 1,282 as shown in Fig 2.13. The bitstrings successfully pass all of these tests with the exception of the NonOverlappingTemplate tests by on average of one to two chips. These tests were performed with the NIST recommended variable values\[55\] applicable for bitstrings of length 1,282.

2.5.2 Reliability

With thresholding, all bit flips are avoided and therefore the intra-chip HD is 0%. The true intra-chip HD is given as 10.819% to illustrate the fraction of the population that is unstable.

| RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES |
|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| C1   | C2   | C3   | C4   | C5   | C6   | C7   | C8   | C9   | C10  | P-VALUE | PROPORTION | STATISTICAL TEST                  |
| 33   | 33   | 33   | 33   | 33   | 33   | 33   | 33   | 33   | 33   | 0.852344 | 32/32 | Frequency                                    |
| 05   | 34   | 25   | 43   | 42   | 34   | 25   | 43   | 42   | 34   | 0.602458 | 32/32 | BlockFrequency                               |
| 24   | 24   | 15   | 54   | 42   | 53   | 34   | 25   | 43   | 42   | 0.739918 | 32/32 | CumulativeSums                               |
| 25   | 04   | 44   | 53   | 32   | 44   | 53   | 32   | 44   | 53   | 0.602458 | 32/32 | CumulativeSums                               |
| 60   | 55   | 24   | 25   | 33   | 33   | 33   | 33   | 33   | 33   | 0.468595 | 32/32 | Runs                                         |
| 16   | 26   | 33   | 31   | 53   | 32   | 32   | 32   | 32   | 32   | 0.299251 | 32/32 | LongestRun                                   |
| 25   | 63   | 52   | 31   | 12   | 3   | 3   | 3   | 3   | 3   | 0.534146 | 32/32 | FFT                                          |
| 32   | 01   | 11   | 01   | 14   | 8   | 8   | 0   | 0   | 0   | 0.000000 * | 32/32 | NonOverlappingTemplate                      |

...  

| C1   | C2   | C3   | C4   | C5   | C6   | C7   | C8   | C9   | C10  | P-VALUE | PROPORTION | STATISTICAL TEST                  |
| 83   | 30   | 03   | 02   | 31   | 2   | 11   | 2   | 0.000010 * | 30/32 | NonOverlappingTemplate |
| 40   | 33   | 54   | 42   | 42   | 45   | 2   | 0.602458 | 32/32 | ApproximateEntropy                             |
| 35   | 36   | 44   | 20   | 43   | 2   | 0.468595 | 31/32 | Serial                                         |
| 44   | 62   | 22   | 14   | 61   | 2   | 0.299251 | 32/32 | Serial                                         |

Figure 2.13: NIST Statistics
2.5.3 Randomness

For a bitstring to be considered random, the first metric considered is whether or not there is a bias towards either 1’s or 0’s. The bitstrings generated produced a ’1’ 50.33% of the time and a ’0’ 49.77% on average (an idea number would be 50% 1’s and 50% 0’s). The difference between the two is well within the ideal margin. This is enough to pass the simple randomness test but isn’t enough for the bitstrings to be considered suitable for cryptography or other applications. Uniqueness is needed such that no bitstring is reproduced from other identical copies.

2.6 Conclusion

In this chapter, I have presented the design and use of an on-chip high resolution measurement structure that provides a method in which 'soft' information can be extracted from a PUF. A thresholding technique for soft information is also presented to increase reliability across industrial temperature and voltage corners. The results show that this measurement structure significantly increases PUF reliability, and resilience to noise introduce by various environmental conditions.
Chapter 3

NotchPUF

3.1 Introduction

Counterfeit electronic components are an increasing concern in the global supply chain of electronic goods. These concerns are reflected in many nations around the world. A 2016 report from the U.S. Chamber of Commerce cites a study conducted by Organization for Economic Co-operation and Development (OECD) estimating that, "global trade-related counterfeiting accounts for 2.5 percent of world trade, or 461 billion [7] U.S. Dollars. This is an increase of 55 percent in less than 10 years. Besides the economic impact, the biggest issue with counterfeit electronic components is the reliability and authenticity of the components being used. Most of the existing work to address these issues utilize Physical Uncloneable Functions (PUFs), counterfeit detection techniques, or cryptographic algorithms in hardware or software built into critical parts of the system.

PUFs derive randomness from the physical characteristics of a device that are relatively easy to measure but difficult to clone. Often, they arise from variations in the manufacturing process that result in unique characteristics of individual devices.
Chapter 3. NotchPUF

PUFs are poised to represent the next generation of hardware security primitives for ICs and fabricated PCBs. The specific identifiers produced by PUFs can serve several applications including unique IDs, authentication, and encryption. PUFs measure and digitize the natural variations that occur in, e.g., path delays, leakage current and SRAM power-up patterns, to produce a long sequence of random bits, i.e., a bitstring. Most of the applications that use these bitstrings require that they be random and unique among the population, and reproducible across adverse environmental conditions. While many of these techniques have matured for manufactured ICs, Field Programmable Gate Arrays (FPGAs), etc., opportunities for improving anti-counterfeiting techniques for PCBs still exist.

To address this board-level security gap, we propose a novel physical unclonable function for PCBs that contains multiple structures to help increase the amount of entropy extracted from a single copper board trace. The proposed solution, called NotchPUF, consists of a number of interdigital capacitor copper trace structures fabricated on the top layer of a 2-layer FR-4 PCB. These structures are designed and tuned to reject 1, 2 and 3 GHz frequencies with various attenuation factors. These frequencies were chosen to prevent aliasing from each series connected structure’s response. Before fabrication, simulations are used to tune each of the three filters individually, and once optimized, additional simulations are used to tune the unified structure with all three filters inserted in series.

The filters are inserted in series to reduce the number of physical I/O driver resources and the amount of time needed for measurement and bitstring generation. To ensure that the PUF design produces bitstrings that are unique among the population, we design the filters’ dimensions to the minimum design rule sizes supported by the fabrication house, thereby maximizing the amount of manufacturing variation effects. The bitstring generation strategy utilizes multiple properties of the measured $S_{21}$ waveform. S-parameters describe the input-output relationship between ports (or
Chapter 3. NotchPUF

terminals) in an electrical system. For instance, in the proposed 2 port system, the symbol $S_{21}$ represents the power transferred from Port 1 to Port 2. We fabricated 48 boards to validate the proposed NotchPUF. The results show that the NotchPUF exhibits high levels of uniqueness and randomness, which are two important statistical properties associated with PUF-generated bitstrings.

This chapter is organized into seven sections. Section 3.2 gives the background on previous related PUF designs and techniques. Section 3.3 presents the NotchPUF design. Section 3.4 shows simulation results to verify the proposed design. Section 3.5 discusses the variations that were observed in the fabricated PCBs. Experimental results in Section 3.6 show the properties, benefits and limitations of the NotchPUF using hardware experiments. Finally Section 3.7 concludes the chapter.

3.2 Related Work

Encryption, authentication, identification and feature activation each utilize random bitstrings as the root component from which the security features of the algorithm and/or protocol are derived. PUFs were introduced in [36], and later refined in [50], to produce unclonable, random and unique keys and bitstrings for these security functions. Since their introduction, a wide range of PUF architectures have been proposed. Each architecture defines a source of entropy, i.e., device-level features that vary randomly because of non-zero manufacturing tolerances. For example, [36] leverages random variations in transistor threshold voltages, [50] uses variations in speckle patterns, [12, 39, 41, 52] measure variations in delay chains and ROs, [15] reads out random power-up patterns in SRAMs, while [21, 31] leverage variations in metal resistance.

As these previous solutions have matured, it has become increasingly apparent that PCBs have many of the same security issues that ICs possess. Similar issues
Chapter 3. NotchPUF

between ICs and PCBs such as counterfeiting and trojan insertion, are described by Ghosh [14]. These authors also present a technique that scrambles the traces on a board, making reverse engineering (RE) more difficult. An obfuscation based approach is described in [17], which uses COTS parts such as a Complex Programmable Logic Device (CPLD) or an FPGA. Their obfuscation approach uses a permutation approach designed to hide inter-chip connections between chips. These types of techniques complicate attempts to reverse engineer layouts to synthesized netlists.

When comparing security issues that exist between both ICs and PCBs, printed circuit boards suffer from a size difference that is several orders of magnitude larger when compared to ICs. Because of this, board features are far more accessible to basic types of manipulations and attacks such as those described by Bhunia and Tehranipoor [6]. Some protections against PCB based attacks are proposed in [11, 27, 48], which physically protect the PCB using an active protection scheme that is able to detect physical tampering. Other protection schemes construct and combine fingerprints of different components on a PCB into a fusion PUF [4]. This fingerprint can be extracted from a variety of components such as a FPGA, SRAM, processor and non-volatile memory. While effective, the components used for PCBs also have supply chain/counterfeit issues. A solution to this issue is proposed in [10] which uses thermal imaging, statistical analysis and machine learning for identification of counterfeit boards and components.

Other authentication techniques have been proposed that fingerprint PCB surface imperfections [28]. Moreover, visual imperfections that occur within PCB vias or through-hole solder pads are used in the fingerprinting process. Although good results are expected during enrollment, a fielded unit may exhibit problems with reproducibility during regeneration if any of the exposed fingerprinted areas suffer from normal “wear and tear” scrapes to the copper plating.

Furthermore, designs have been proposed that analyze variations within passive
components [18], debug components [22], or a combination of passives that define a filter structure [53]. While surface mount components are plentiful on any modern PCB design, measuring each individual component to create a unique identifier becomes impractical and costly. Another element present on all PCBs are copper traces. The variations that exist within copper traces have been used as the foundation of unique identifiers [18, 68, 71, 73]. These techniques measure variations in single and double layer copper traces that are replicated across the PCB at different locations [18, 73]. Double layer copper traces are constructed as transmission lines and used with several PCB elements to check the integrity of the PCB. These elements have embedded PUFs that are capable of generating individual authentication IDs.

A type of device that is commonly utilized in association with Radio Frequency (RF) equipment is the notch filter (aka band reject/stop filter). These filters are typically composed of a low-pass and high-pass filter that are connected in a parallel configuration. The notch filter device enables passage of specific frequencies and rejects others. It is also called band elimination filter or notch filter. In 1970, Alley [2] proposed interdigital capacitors for use in lumped-element microwave ICs. The microstrip interdigital resonator is equivalent to the planar interdigital capacitor, whose operating frequency can be changed by controlling the capacitance. The microstrip notch filter uses open circuited stubs consisting of one main transmission line coupled to a half-wavelength resonator, which is typically electrically and magnetically coupled to it. The resonance frequency depends on the half-wavelength resonator. However, open-circuited stubs are large and not well suited as a PCB PUF. The microstrip notch filter with shunt stubs of a quarter wavelength must be larger than the quarter-wavelength resonator [25]. To increase the number of PUF structures that can exist on a PCB, a compact solution is needed. A number of structures have been proposed that show the decrease in size that PCB resonators have undergone over the years [1, 44, 69].
Chapter 3. NotchPUF

IC PUFs have continued to mature while the need for a reliable PCB based PUF still exists. Current work in the PUF field has laid the ground work for more complex structures such as those introduced here. Besides trace impedance variations, there exists many additional sources of entropy in a PCB that have yet to be fully utilized. Motivated by the above, we propose the NotchPUF, a set of tuned microstrip filter PCB structures that leverage this entropy for the creation of a unique board identifier.

3.3 Interdigital Notch Filter

The goal of the NotchPUF is to leverage random manufacturing variations that exist within a sequence of microstrip resonator structures to create a unique, unclonable identifier for the PCB which can be used for authentication. In this section, we describe the structural and electrical characteristics of the proposed NotchPUF.

Figure 3.1: Interdigital Resonator with Equivalent circuit
Chapter 3. NotchPUF

3.3.1 Interdigital Resonator

This interdigital resonator consists of two comb microstrip structures directed opposite of each other, forming thereby the so-called interdigital structure. The common conductor of each comb structure are considered Ports 1 and 2. Each of the fingers in the structure can be considered itself as a quarter-wave resonator. The initial structure of this resonator is shown in Fig. 3.1. Here, \( L_1 \) represents the added inductance from the connected Port 1 and 2. \( C_p \) and \( C_g \) the parallel capacitance between each ports respective fingers, and finally \( C_1 \) the capacitance for the comb [47].

3.3.2 Double Interdigital Resonator

![PCB structure of a single notch filter with a double microstrip interdigital resonator](image)

Figure 3.2: PCB structure of a single notch filter with a double microstrip interdigital resonator

The NotchPUF consists of three structures that each contain a double interdigital resonator, which can be tuned to reject specific frequencies. A double interdigital resonator was chosen to increase the amount of attenuation the structure has. This
Chapter 3. NotchPUF

will increase the amount of sensitivity at their designed resonant frequency. Given the high-frequency nature of RF circuits, these resonators are inherently sensitive to various physical parameters of the PCB such as trace widths, spacing, board thickness, substrate dielectric constant, etc. The double resonator structure is shown in Fig 3.2. The input port (Port 1) and output port (Port 2) are connected via two resonators and a length of copper (W3), to form the notch filter structure. Since each resonator has more than two fingers, the interdigital capacitance between each finger must be considered. Equation (3.1) from [2] can be used to estimate the interdigital capacitance for a single resonator structure.

\[
C(\rho F) = \frac{\varepsilon_r + 1}{W_1} l_2(\varepsilon_r + 1)[0.1(n - 3) + 0.11] 
\]

Here, \(n\) is the number of fingers (6 in this case), and \(W_1\) is the line width (3mm) of the microstrip interdigital resonator, as shown in Fig. 3.2. The interdigital capacitance affects the center frequency of the designed filter, which classifies the interdigital notch filter as a semi-lumped element device. Furthermore, the center frequency of the filter can be tuned by varying the number of fingers (n) and the gap spacing S1. Using (3.1), we were able to derive an accurate capacitance value that enabled each of the filter structures to be tuned to a specific rejection frequency.

3.3.3 NotchPUF

The proposed notch filter is designed to increase the level of randomness that can be extracted over that of a single copper trace line by amplifying small PCB manufacturing variations, in particular, those associated with the interdigital capacitor. In contrast, single copper trace wires that traverse the PCB can suffer from an “averaging effect” where small board-level variations in wire widths and capacitances
Chapter 3. NotchPUF

Figure 3.3: PCB structure of a single NotchPUF

are averaged out over the length of the trace, effectively reducing the magnitude of signal variations that can be measured and leveraged by the PUF.

To reduce the area overhead associated with the proposed NotchPUF, we connect 3 notch filter elements in series. The three filters have similar physical characteristics but are distinct in their rejection frequencies. For example, each notch filter has the same size traces, the same distance between fingers, the same 50 ohm characteristic trace impedance, and only the length L1 is varied among the 3 copies to achieve different rejection frequencies. A full-wave Electromagnetic (EM) simulator was used to design and tune the proposed notch filter with interdigital resonator center frequencies of 1, 2, and 3 GHz. These filters are then combined in series, and re-simulated to further tune the response of the individual filters. The process used for tuning each these structures is discussed in more detail in Section 3.4. The final structure of the NotchPUF is shown in Fig. 3.3.
3.4 Simulations

Full-wave EM simulations were run to determine the optimal size and spacing of the filter’s physical geometries to achieve the target notch frequencies of 1, 2 and 3 GHz. Each of the three notch filters were simulated individually while varying the parameters shown in Fig. 3.2 across a range of values. Additional details of these simulations are provided in the following.

3.4.1 Tuning

In CST Studio [62], a single double-resonator was modeled on a FR-4 substrate with a thickness of 1.6 mm. FR-4 is a low-cost printed circuit board material, manufactured from fiberglass cloth embedded within an epoxy resin. A full copper ground plane is placed on the entire bottom layer of the board with a 1 oz copper pour (1.4 mils). Ports 1 and 2 used discrete ports from each port to the ground plane. This type of port was used due to the small separation between the input ports. This resonator is shown in Fig 3.4.

Figure 3.4: Double Interdigital Resonator
Chapter 3. NotchPUF

Using simulations, the length parameters $L_1$, $L_2$, $S_1$, $W_2$, and $G_1$ (Fig. 3.2) were tuned until the double interdigital resonator had a center frequency of 1 GHz. Once the optimal parameters for the 1 GHz structure were determined, only $L_1$ was changed to get center frequencies of 2 and 3 GHz. Fig. 3.5 illustrates the impact of using several different values of parameter $L_1$ on the center frequency of the double microstrip resonator. From the plot, decreasing $L_1$ causes the center frequency to increase. Moreover, because the other length parameters are held constant, and not simultaneously optimized in these simulation results, the Fractional Bandwidth (FBW) percentage also increases (filter gets wider).

![Figure 3.5: $S_{21}$ Notch filter tuning](image)

The simulation process that we used tuned all parameters simultaneously to achieve the optimal result, which yielded the following values: $L_{1(1GHz)} = 23.30$ mm, $L_{1(2GHz)} = 14.50$ mm, $L_{1(3GHz)} = 11.50$ mm, $L_2 = 8.35$ mm, $L_3 = 3.0$ mm, $W_1 = 3.0$ mm, $W_2 = 0.20$ mm, $W_3 = 1.0$ mm, $G_1 = 0.40$ mm and $S_1 = 0.15$ mm. As noted earlier, only $L_1$ is different in the 3 versions of the filter.
3.4.2 NotchPUF Simulations and Validation in Hardware

The individually tuned double-resonator structures were then connected in series to define the proposed structure of the NotchPUF as shown in Fig. 3.3. Additional simulations were used to fine tune the parameters of the unified NotchPUF structure. Fig. 3.6 plots the simulated performance and actual performance of a fabricated NotchPUF [PCB] together. Table 3.1 gives the values of several important response characteristics of the NotchPUF obtained from the simulations.

<table>
<thead>
<tr>
<th>Structure</th>
<th>$f_c$ (GHz)</th>
<th>FBW</th>
<th>IL (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 GHz</td>
<td>1.008</td>
<td>15.9%</td>
<td>-26.91</td>
</tr>
<tr>
<td>2 GHz</td>
<td>2.028</td>
<td>17.8%</td>
<td>-40.68</td>
</tr>
<tr>
<td>3 GHz</td>
<td>3.012</td>
<td>13.6%</td>
<td>-43.09</td>
</tr>
</tbody>
</table>

Table 3.1: Simulation Properties

Here, $f_c$ is the central frequency, -3dB for Fractional Bandwidth (FBW), and Insertion Loss (IL). Equation (3.2) gives the expression to compute the fractional bandwidth.

$$FBW = (f_2 - f_1)/f_c$$  \hspace{1cm} (3.2)

Here, $f_1$ is the lower frequency, $f_2$ is the upper frequency and $f_c$ is the center frequency. Although differences between the measured and simulated frequencies can be attributed to modeling errors, we show in the following that a large fraction of this difference is actually introduced by variations in the fabrication process.


### 3.5 PCB Variation

The NotchPUF leverages shifts in the rejection frequency and changes in the attenuation at the rejection frequency as a source of entropy. Variations that occur in these response characteristics across PCBs are rooted in the manufacturing process. PCBs are fabricated by decomposing the elements in the design, wires and vias, into a sequence of layers. For example, a two layer board consists of a top copper layer, a middle dielectric substrate layer (usually woven glass and epoxy) and a bottom copper layer. The copper and substrate layers are bonded together using a lamination process that applies heat and pressure to the three layers. Traces on the copper layers are created using a mask and high intensity UV light. The unexposed regions are then etched with a chemical solution. Conducting vias between the layers are created using a drilling and plating process. PCBs with additional layers are created by applying a dielectric prepreg bonding layer to the two layer board and then another etched copper layer, with drilling and plating as needed. The outside
Chapter 3. NotchPUF

layers are finished by using a solder mask and tinning process. Additional details can be found in [34].

All of these PCB processing steps are subject to manufacturing variations. Moreover, the magnitude of the variation is directly related to the size of the drawn component, with smaller features typically exhibiting larger levels of variations. We designed the geometries of the NotchPUF to be close to the minimum feature size allowed by the fabrication house as a means of increasing the probability that different boards will produce different responses. We used the PCB board manufacturer specified tolerances to guide our selection of certain design parameters. For example, the manufacturer specified tolerances on wire width and spacing are +/- 20% [64]. In order to maximize the impact of these tolerances on the filters’ response characteristics, we designed the metal wires within the notch filter to be the minimum trace width and spacing of approx 7 mils. Other design-independent tolerances, such as +/- 10% on board thickness, also impact the filters’ response characteristics.

The NotchPUF is composed of a microstrip transmission line with series-inserted filter elements. Microstrip transmission lines posses a characteristic impedance that can be modeled by Equations 3.3 through 3.8 taken from [67].

For $\frac{W}{h} < 1$:

$$\varepsilon_{\text{eff}} = \frac{\varepsilon_{x}+1.0}{2} + \frac{\varepsilon_{x}-1.0}{2} \left[ \frac{1}{\sqrt{1+\frac{h}{W}}} + 0.04 \left( 1 - \frac{W}{h} \right)^2 \right]$$  \hspace{1cm} (3.3)

Else:

$$\varepsilon_{\text{eff}} = \frac{\varepsilon_{x}+1.0}{2} + \frac{\varepsilon_{x}-1.0}{2} \left[ \frac{1}{\sqrt{1+\frac{h}{W}}} \right]$$  \hspace{1cm} (3.4)
Chapter 3. NotchPUF

And:

\[ Z_0 = \frac{120\pi}{2\sqrt{2\pi} \sqrt{\varepsilon_r + 1}} \]

\[
\ln \left\{ 1 + \frac{4h}{W'} \left[ \frac{14 + \frac{8}{\varepsilon_{eff}}}{11} \frac{4h}{W'} \right] \right. \\
+ \left. \sqrt{\left( \frac{14 + \frac{8}{\varepsilon_{eff}}}{11} \right)^2 \left( \frac{4h}{W'} \right)^2 + \frac{1 + \frac{1}{\varepsilon_{eff}}}{2} \pi^2} \right\} 
\]

(3.5)

Where:

\[ W' = W + \Delta W' \] (3.6)

\[ \Delta W' = \Delta W \left( \frac{1.0 + \frac{1.0}{\varepsilon_{eff}}}{2} \right) \] (3.7)

\[ \Delta W = \frac{t}{\pi} \ln \left[ \frac{4e}{(t/h)^2 + \left( \frac{1}{w/t+1.1} \right)^2} \right] \] (3.8)

Here, \( \varepsilon_r \) is the relative dielectric constant, \( W \) is the width of the copper trace, \( t \) is the thickness of the copper track, and \( h \) is the thickness of the dielectric substrate. From these equations, we investigate variations in the following parameters:

1. Board thickness
2. Trace width
3. Substrate relative dielectric constant

While not a comprehensive list of parameters that exhibit variations in the PCB manufacturing process, these particular parameters have the largest tolerance levels, and therefore, are the most difficult to precisely control during manufacturing.

To better understand the impact of these parameters on the filters’ response characteristics, we fabricated a set of 46 2-layer FR-4 boards and measured the actual
board thickness and trace widths directly. In a second experiment, we fabricated another set of 2-layer FR-4 boards and measured the variations in the relative dielectric constant of the substrate. It should be noted that although two different test boards are used in our measurements, all PCBs came from the same 18 inch by 24 inch panel used as the source material in the manufacturing process.

3.5.1 Board Thickness Variations

![Figure 3.7: NotchPUF Board Measurement Locations](image)

To quantify the amount of variation in the fabricated boards, 10 board thickness measurements were taken at each of the 3 locations on each of the NotchPUF boards as shown in Fig. 3.7. The position indicated with an “X” was not used because a board label on the bottom-side of the board (not shown) adds to the thickness and would skew the results. The ground plane on the bottom-side of the board is added using a 1 oz copper layer process, which is nominally 1.4 mils in thickness. The manufacturer specifies the nominal thickness of the FR-4 substrate as 1.6 mm (or 63 mils). In addition to these two layers, the solder mask on each side of the PCB also
Chapter 3. NotchPUF

adds to the thickness. Although not specified by the manufacturer, these layers are typically approximately 0.8 mils to the thickness. Therefore, the nominal thickness of the PCB is 1.68 mm (66 mils) when all layers are considered.

Each position was measured with a calibrated micrometer with accuracy of 0.0001”. The three measurements from each board were averaged to give the overall thickness per board. The results of these measurements are shown in Fig. 3.8. The average board thickness of all boards combined is 1.5375 mm which differs from the nominal value by approximately 9%. Although the thickness are smaller than the expected 1.68 mm (66 mils), they are still within the 20% allowable manufacturing tolerance.

![NotchPUF Board Thickness Variation, 3σ* = 0.11439](image)

**Figure 3.8: Board Thickness Graph**

### 3.5.2 Trace Width Variation

To analyze trace width variations, we used an image processing approach where the 3 filter structures were imaged under a microscope. All of the images were aligned via Matlab image processing code, and then the raw files were converted to gray-scale. The gray-scale images were processed with a tuned ‘canny’ edge detection filter that
produced a low noise delineation of the copper traces. The widths of the traces were measured by counting the number of pixels between opposing edges.

The length of a pixel was determined using a microscope calibration slide with 0.01 mm line resolution. The calibration slide was imaged with the same zoom and focus levels that were used to image the NotchPUF structures. The measurements were divided by the pixel widths of the calibration sources to derive an estimate of the pixel length. From these measurements, each pixel was determined to be .002217 mm (or 2.217 µm).

The distribution of trace widths obtained from these measurements are shown in Fig. 3.9. As noted earlier in reference to W2 in Fig. 3.3, the designed finger trace widths is 0.2 mm. The title bar indicates that the 3σ variation of the trace widths is approximately 0.0302 mm. The average trace width (not shown) is 0.1930 mm, which is slightly smaller than the design width of 0.2 mm. These results show that the finger trace widths vary by approx. 15%, which are again within the manufacturing tolerance of 20%. This same amount of variation (0.03mm) was observed on
both lengths \( L_3 \) and \( W_1 \). From simulations, this variation was determined to have negligible effect on the NotchPUF response when compared to the finger width.

### 3.5.3 Substrate Relative Dielectric Constant Variation

![Figure 3.10: Two Microstrip Dielectric Constant Experiment](image)

To determine the substrate relative dielectric variation, measurements were taken on a different set of 20 PCBs which included a specialized trace configuration. On this PCB, two 50-\( \Omega \) microstrip transmission lines are copper etched with lengths 50 mm and 100 mm, respectively. A photo of this board is shown in Fig. 3.10. We use an Agilent 8753E Vector Network Analyzer (VNA) to measure the difference between the electrical lengths of the two microstrips. The VNA applies a sequence of sine wave signals at specific frequencies to the input port and measures the response on the output port. The responses are then analyzed to derive the effective dielectric constant \( \varepsilon_{eff} \) of the microstrip structure as a function of frequency using Eqs. 3.9 and 3.11. For a microstrip structure, part of the electric field is in air and a part is in
Chapter 3. NotchPUF

the dielectric. If the dielectric constant of the material is $\epsilon_r$, the effective dielectric constant $\varepsilon_{\text{eff}}$ must be less than $\epsilon_r$, i.e., between 1 and $\epsilon_r$, because part of the electric field is in air and the dielectric constant of air is 1 [61].

$$\Delta \phi = 2\pi f (\Delta l_p) \frac{\sqrt{\varepsilon_{\text{eff}}}}{c}$$

(3.9)

Where:

$$\Delta l_p = l_{p1} - l_{p2}$$

(3.10)

And

$$\Delta l_e = \sqrt{\varepsilon_{\text{eff}} \ast \Delta l_p}$$

(3.11)

In these equations, $\Delta \phi$ is the transfer phase difference through the two lines at a particular frequency, $\Delta l_p$ is the difference between the physical lengths and $\Delta l_e$ is the electrical lengths of the lines. Our test board setup shown in Fig. 3.10 which includes two microstrip lines of different lengths allows the impact of the SubMiniature version A (SMA) connectors on the response to be removed because all 4 cable-to-microstrip transitions are identical. This is achieved through subtraction when computing $\Delta l_e$ in Eq. 3.11 [9]. This method has been shown to be capable of accurately measuring the dielectric constant of a substrate with an error margin between 0.5% and 1.0% [9]. We observed a variation of approx. 4.7% in the effective dielectric constant of the microstrip structure. Table 3.2 gives the average $\Delta \phi$ and $\epsilon_{\text{eff}}$ computed using data from all 20 PCBs for a range of frequencies between 1.0 and 4.0 GHz. The 3$\sigma$ for $\epsilon_{\text{eff}}$ is shown in the right-most column.

From these measurements, we were then able to determine $\varepsilon_r$ of the substrate around the test frequency using an analysis of the microstrip propagation constant.
Table 3.2: Average Measurement of $\Delta \phi$ (degrees) and $\varepsilon_{eff}$ Variations in FR-4 ($\varepsilon_r=4.8$) Substrate

<table>
<thead>
<tr>
<th>f(GHz)</th>
<th>$\Delta \phi$ (measured)</th>
<th>$\varepsilon_{eff}$ (calculated)</th>
<th>$3\sigma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>114.59°</td>
<td>3.63</td>
<td>0.17</td>
</tr>
<tr>
<td>1.5</td>
<td>174.15°</td>
<td>3.73</td>
<td>0.17</td>
</tr>
<tr>
<td>2.0</td>
<td>231.50°</td>
<td>3.72</td>
<td>0.17</td>
</tr>
<tr>
<td>2.5</td>
<td>288.13°</td>
<td>3.68</td>
<td>0.18</td>
</tr>
<tr>
<td>3.0</td>
<td>345.10°</td>
<td>3.67</td>
<td>0.19</td>
</tr>
<tr>
<td>3.5</td>
<td>400.03°</td>
<td>3.62</td>
<td>0.15</td>
</tr>
<tr>
<td>4.0</td>
<td>463.06°</td>
<td>3.71</td>
<td>0.22</td>
</tr>
</tbody>
</table>

The following system of equations can be solved to derive the relative dielectric constant of the FR4 substrate: \[19\].

$$\varepsilon_{eff}(u, \varepsilon_r) = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2}(1 + \frac{10}{u})^{-a(u)b(\varepsilon_r)}$$ \hspace{1cm} (3.12)

$$a(u) = 1 + \frac{1}{49} ln \left( \frac{u^4 + (u/52)^2}{u^4 + 0.432} \right) + \frac{1}{18.7} ln \left[ 1 + \left( \frac{u}{18.1} \right)^3 \right]$$ \hspace{1cm} (3.13)

$$b(\varepsilon_r) = 0.564(\frac{\varepsilon_r^{-0.9}}{\varepsilon_r + 3})^{0.053}$$ \hspace{1cm} (3.14)

$$u = \frac{w}{h}$$ \hspace{1cm} (3.15)

Here, $u$ is the strip width normalized with respect to substrate height. We used Matlab \[43\] to iteratively solve this system of equations for different values of $\varepsilon_r$ until a match occurred to the measured values $\varepsilon_{eff}$. 

52
Chapter 3. NotchPUF

The calculated \( \varepsilon_r \) was then analyzed and compared between all 20 boards and shown to have approx 5.3% variation amongst the population of PCBs. \( \varepsilon_r \) was then averaged to indicate the expected dielectric constant for the substrate of the entire PCB panel. These are summarized in Table 3.3 below.

<table>
<thead>
<tr>
<th>f(GHz)</th>
<th>( \varepsilon_r ) (calculated)</th>
<th>3( \sigma )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>4.86</td>
<td>0.26</td>
</tr>
<tr>
<td>1.5</td>
<td>5.02</td>
<td>0.25</td>
</tr>
<tr>
<td>2.0</td>
<td>4.99</td>
<td>0.25</td>
</tr>
<tr>
<td>2.5</td>
<td>4.93</td>
<td>0.27</td>
</tr>
<tr>
<td>3.0</td>
<td>4.91</td>
<td>0.27</td>
</tr>
<tr>
<td>3.5</td>
<td>4.85</td>
<td>0.22</td>
</tr>
<tr>
<td>4.0</td>
<td>4.98</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Table 3.3: Variation of \( \varepsilon_r \) between boards in FR-4 (\( \varepsilon_r = 4.8 \)) Substrate

3.5.4 Summary

The previous three sections provide evidence that manufacturing variations in PCBs are significant, and therefore, can serve as a source of randomness for PUFs. We summarize our findings as follows:

**Board Thickness:** The average board thickness (with the bottom copper layer) was shown to be 1.5375 mm which is approximately 9.2% thinner than the manufacturer specified thickness of 1.68 mm. More importantly, the average 3\( \sigma \) variation across boards was measured to be 0.1143 mm or approximately 7.4%. Although relatively small in comparison to other sources of variations, these board thickness variations impact the characteristic impedance of the microstrip line within the Notch-PUF, and therefore add to the variations in the measured response discussed below in the experimental results.

**Trace Width Variation:** The average width of a trace was determined to be
Chapter 3. NotchPUF

approximately 0.1930 mm with a $3\sigma = 0.0302$ mm. Although the average differs by only 3.6% from the drawn width, the $3\sigma$ variation is much larger at approximately 15%. Similar to board thickness variations, trace width variations impact the characteristic impedance of the transmission line. As we show in the next section, trace width variations have a significant impact on the center frequencies of the notch filters.

**Substrate Relative Dielectric Variation:** We computed the average value for $\varepsilon_r$ at all frequencies to be approx. 4.93, which is approximately 2.7% greater than the manufacturer claimed value of 4.8. The mean $3\sigma$ was computed to be 0.26, which represents a 5.3% variation. Although the level of variations here are smaller than the former two, dielectric variations also impact characteristic impedance, and add to the randomness we observe in the center frequency and magnitude of the response characteristics of the NotchPUF.

3.6 Experimental Results

3.6.1 Experimental Setup

The proposed NotchPUF is fabricated on a 50 mm by 50 mm, 2-layer FR-4 PCB with a thickness of 1.6 mm and a relative permittivity $\varepsilon_r$ of 4.8. We fabricated 46 boards and used an Agilent 8753E VNA and an 85047A S-Parameter test set to make the measurements. Because the VNA is limited to 1601 points of data for a given center frequency and span, three separate measurements were taken at frequency spans of 1 GHz, with center frequencies set to 1, 2 and 3 GHz. At each center frequency, we repeated the testing to obtain 200 samples. A set of 20 sample averages were computed from this raw data by averaging 10 consecutive measurements. Fig. 3.11 shows the instrumentation setup with the Agilent VNA.
Chapter 3. NotchPUF

tasked with data collection and the HP 85047A applying the S-Parameter test set to
the NotchPUF PCB connected between Ports 1 and 2.

Prior to data collection, a full 2-port calibration was performed on the VNA with
an open, load and short calibration standard suite of tests. Calibration is designed
to reduce instrumentation error including directivity error, crosstalk, source match
error, frequency response reflection tracking error, and frequency response transmis-
sion tracking error. Calibration records data within the instrument to address a total
of twelve error terms, six associated with the forward direction tests and six with
the reverse direction tests. The same phase stable cable was used during calibration
and NotchPUF testing, which further reduced instrumentation-related noise. The
2 SMA connectors on each board were also analyzed to ensure minimal impact to
the NotchPUF measurements. All connectors were validated to have a worst case
insertion loss $SMA_{IL}$ as shown in eq (3.16)

$$SMA_{IL} = 0.05\sqrt{f(GHz)} \ dB.$$  \hspace{1cm} (3.16)
Chapter 3. NotchPUF

After calibration, $S_{21}$ measurements were taken from each board and post-processed to extract a set of response characteristics including center rejection frequency, maximum insertion loss (lowest -dB), $\sigma_{\text{FBW}}$ and points of maximum roll-off (slope of curve). The values of these parameters were then combined with the frequency and insertion loss after applying a modulus technique similar to that in [8,31]. We use a modulus operation to ensure an unbiased response bitstring, defined here as a 15-bit ID for each NotchPUF PCB as we explain further below.

The statistical quality of the bitstrings produced by NotchPUF are evaluated using several statistical metrics. In particular, we compute inter-board HD which measures the uniqueness of the ID$^k$. We also compute intra-board HD using bitstrings generated by the same NotchPUF PCB to determine how well each NotchPUF is able to reproduce the bitstring. We compare the computed statistics with the ideal values of 50% for uniqueness and 0% for reproducibility.

3.6.2 Signal Analysis and Bitstring Generation

First, is the average frequency and $S_{21}$ insertion loss generated by each NotchPUF PCB. The center frequency and the insertion loss are plotted as separate curves with one point for each of the 46 boards identified along the x-axis in Fig. 3.12. The rejection frequencies for the PCBs varies above and below the target rejection frequency, e.g., 1 GHz as shown on the y-axis in Fig. 3.12(a). The results of the NotchPUF measurements are summarized in table 3.4 below. The average center rejection frequency for each filter represents approximately a 4.4% deviation from the target frequencies of 1, 2 and 3 GHz. The $\sigma_{\text{FBW}}$ for each structure is 67.99 MHz, 94.73 MHz, and 187.75 MHz. The variations that occur in each of these response parameters are used to define a 15-bit identifier for each PCB discussed in section 3.6.3.
Chapter 3. NotchPUF

(a) 1 GHz Structure
(b) 2 GHz Structure
(c) 3 GHz Structure

Figure 3.12: Frequency and Insertion Loss for NotchPUF PCBs

<table>
<thead>
<tr>
<th></th>
<th>1GHz</th>
<th>2GHz</th>
<th>3GHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Center rejection frequency</td>
<td>1.034 GHz</td>
<td>2.072 GHz</td>
<td>3.134 GHz</td>
</tr>
<tr>
<td>Rejection Frequency $3\sigma$</td>
<td>59.18 MHz</td>
<td>118.4 MHz</td>
<td>192.3 MHz</td>
</tr>
<tr>
<td>Insertion Loss $3\sigma$</td>
<td>7.87 dB</td>
<td>17.92 dB</td>
<td>23.27 dB</td>
</tr>
<tr>
<td>FBW</td>
<td>304.64 MHz</td>
<td>561.67 MHz</td>
<td>414.54 MHz</td>
</tr>
</tbody>
</table>

Table 3.4: NotchPUF Parameter Results
We create a graph in Fig. 3.13 which plots insertion loss (y-axis) against frequency (x-axis) for each of the 46 boards as a means of converting the measured values into a bitstring. The data shown here is for the 2 GHz filters, but similar graphs can be created using the 1 and 3 GHz data. Each PCB is associated with one cluster of points in the graph, where the points in the clusters represent the 20 sample averages discussed earlier. The sample averages allow the noise levels associated with the measurements to be compared directly to magnitude of the variations that occur between PCBs in the population.

The variations that occur in the filters’ response are very large, which enables more than one bit to be obtained from the measurements. The data plotted in Fig. 3.13 is reduced in amplitude over the original values by applying a modulus operation, which effectively wraps the values into a smaller region. The modulus operation accomplishes two goals. First, it translates the (x,y) points associated
with each PCB into a fixed size region. In our example, the region is bounded by 0 to 40 MHz along the x-axis and -10 to 0 along the y-axis. The fixed size region in turn enables boundaries to be defined for the assignment of bit values, as shown by the four quadrants in the figure. Second, the modulus removes bias that occurs across PCBs but preserves the within-PCB variations, which improves randomness in the generated bitstrings.

From Fig. 3.13, there are cases in which the cluster of points for each PCB cross over a boundary. When this occurs during regeneration in the field, bit-flip errors will occur in the bitstrings. Although the bit-flip errors occur in our experiments because of measurement noise, we also expect this to occur if the temperature of the environment changes. In either case, we propose to use error correcting techniques to deal with bit-flip errors as others have done in previous work [70].

Applying this process to the data from the three filters on each PCB generates six bits of the PCBs ID. Three additional bits are obtained by processing the FBW data, and another six bits using the maximum rolloff variation on each side of the $S_{21}$ response. Therefore, a total of 15 bits can be generated for each PCB using the three filters.

A bitstring of size 80-bits is traditionally considered the smallest acceptable size for authentication operations. The size of the structure containing all three filters as shown in Fig. 3.7 is 31 mm x 23.3 mm. A 2 x 3 array of the NotchPUF structures would be able to generate 90 bits and would occupy a board area of 70 mm x 62 mm, less than a 3”x2.5” area of board real estate. Although 18 notch filter structures are included in this array, the number of required inputs and outputs is only 6 because of the series connection strategy.

We would also like to point out that the NotchPUF does not need to be implemented on the top or bottom layers of the PCB as we have done here. In fact, a
better location would be to embed them in a buried layer within a multi-layer PCB. A buried implementation would increase the difficulty of physical tampering. Moreover, the widths of the notch filters’ wires, and overall size, would decrease because of their close proximity to adjacent ground planes.

### 3.6.3 Bitstring Analysis

The uniqueness and reliability characteristics of the 15-bit bitstrings from the 46 boards were evaluated using inter-PCB and intra-PCB Hamming Distance (HD). The HD between two n-bit bitstrings x and y is defined as

\[
HD(x, y) = \sum_{i=1}^{n} (x_i \oplus y_i)
\]

Fig. 3.14 plots the inter-PCB hamming distance distribution while 3.15 plots intra-PCB distribution. To increase statistical significance, the distributions were created using all 200 measurements taken per NotchPUF PCB. The mean inter-PCB HD from Fig. 3.14 is 48.7%, which is close to the ideal value 50%. From 3.15, the intra-PCB HD is computed as 2.8%, which is well within the error correction capabilities of error correction methods. We observed larger numbers of bit-flip errors for the 3 GHz filter, suggesting that filters with higher rejection frequencies are more sensitive to noise.

FR-4 is an inexpensive board material and is therefore very popular. However, the signal losses are much higher than they are for more specialized (and expensive) board materials, such as polytetrafluoroethylene (PTFE), which has excellent dielectric properties at microwave frequencies. However, we expect FR-4 will be the material of choice in most cases, and therefore recommend that notch filter designs be constrained to operate below 3 GHz for repeatability.
Chapter 3. NotchPUF

The experiments here were only performed at nominal temperature (25°C). As ambient temperature increases, the FR-4 PCB dielectric constant increases approximately 4% from -30°C to 105°C [20]. This change in \( \varepsilon_r \) results in a corresponding linear change to the center rejection frequencies of the NotchPUF. Simulation results illustrating the temperature effect on \( \varepsilon_r \) are shown in Fig. 3.16. The change in center frequency is proportional to the percentage change of the relative dielectric change (5%). For example, the value of \( \varepsilon_r \) is 4.3 at room temperature (25°C) and is 4.6 at 105°C. The shift in \( \varepsilon_r \) introduced by changes in temperature can be calibrated by offsetting the measured center frequency. However, this requires that the PCB have components capable of measuring the ambient temperature or that the PCB is able to store a set of reference “room temperature” values for each filter in a non-volatile memory.

3.7 Conclusions

In this chapter, a novel PUF called NotchPUF is proposed that utilizes PCB variations in notch filter structures for the creation of unique IDs for PCBs. The proposed design creates a series connected set of notch filters with different rejection frequen-
Chapter 3. NotchPUF

Figure 3.16: Changes in Er with Temperature

...cies to reduce the area overhead of the PUF architecture. Simulation experiments are used to tune wire sizes and other parameters associated with a set of three filter designs with target rejection frequencies of 1, 2 and 3 GHz. Fabricated PCBs are designed and tested to measure the magnitude of the variations associated with the fundamental elements of PCBs including wire width, substrate thickness and dielectric constants. Data from the PCBs are processed into bitstrings using a novel quadrant-based modulus scheme, and the bitstrings from 46 PCBs are analyzed using standard statistical bitstring quality tests to illustrate that high levels of randomness and uniqueness can be achieved from the NotchPUFs in our proof-of-concept experiments.
Chapter 4

Multi-panel NotchPUF PCB Variation Analysis

While the previous results were very promising for the 48 2-layer PCB boards that were created, the question of how the results compared between new sets of boards arose. To further understand the variation that exists in a 2-layer PCBs, experiments detailed in section 3.5 were repeated. This time, data from 4 panels of PCBs were collected. A total of 174 boards were considered, with 46 PCBs from panel 1, 44 from panel 2, 44 from panel 3, and finally 40 from panel 4. In order to increase the throughput of printed circuit board (PCB) manufacturing, PCBs are often designed so that they consist of many smaller individual PCBs that will be used in the final product. This PCB cluster is called a panel or multiblock. The large panel is broken up or “depaneled” as a certain step in the process, after in-circuit test (ICT), after soldering of through-hole elements, or even right before the final case-up of the assembly.

Each PCB panel was created using the same exact layout design to ensure no skew of results and allow for detailed comparison. Each separate PCB panel was
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

created with a different color laminate to help visually differentiate the originating panel. Variation results from section 3.5 used red PCBs and the data obtained from those results are re-used in this chapter and have a board designation of “*H1”. Boards labeled “*H2”, “*H3” and “*H4” designate the 3 other panels that the PCBs originated. The “*” preceding each panel designates the individual board number such as Board 1 Panel 1 (B1H1). These various PCBs and corresponding board labels is shown in Fig 4.1.

Figure 4.1: All PCB Boards Considered

This chapter is formatted as follows: Section 4.1 discusses the variations that were observed from the 4 different panels of PCBs. Section 4.2 discusses the experimental results of the NotchPUF in relation to the panel it originated from. Challenges that were encountered in 4.3. Finally, the conclusions are discussed in section 4.4.


Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

4.1 PCB Variation

The first step in realizing the usefulness of a NotchPUF like the NotchPUF, is to understand how PCBs might differ between batches. Variations that exist between each panel is often proprietary information known only to the producing factory and often needs a non-disclosure agreement (NDA) signed before any information is released [51]. To account for defects, factories quite often tune parameters while production continues to run [33]. Because of this, the previous designed experiments are re-used, and the results used to determine overall variation between panels of a PCB.

4.1.1 Board Thickness

Recalling previously, 10 board thickness measurements were taken at each of the 3 locations on each of the NotchPUF boards designated by a number shown previously on Fig. 3.7. The position indicated with an “X” was not used because the label on the bottom-side of the board would add to the thickness and would skew the results. The ground plane on the bottom-side of the board uses a 1 oz copper pour, which is nominally 1.4 mils in thickness. The manufacturer specifies the nominal thickness of the FR-4 substrate as 1.6 mm (or 63 mils). In addition to these two layers, the solder mask on each side of the PCB also adds to the thickness. Although not specified by the manufacturer, these layers are typically approximately 0.8 mils to the thickness. Therefore, the nominal thickness of the PCB is 1.68 mm (66 mils) when all layers are considered. These results are shown in Fig 4.2.

Here, each colored line designates a different panel of PCBs, with board number on the x-axis and the measured thickness on the y-axis. The board thickness stated from the manufacturer is indicated by the dotted red line located at the calculated 1.68mm. PCB panel H1 appears to vary the most with a $3\sigma$ variation of 0.1144mm.
Panels 2,3 and 4 appear to have roughly the same amount of variation with the $3\sigma$ and average board thickness as shown in Table 4.1.

Table 4.1: PCB Panel Board Thickness Results

<table>
<thead>
<tr>
<th>Panel</th>
<th>Avg Thickness (mm)</th>
<th>$3\sigma$ (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>H1</td>
<td>1.5375</td>
<td>0.1144</td>
</tr>
<tr>
<td>H2</td>
<td>1.5982</td>
<td>0.0298</td>
</tr>
<tr>
<td>H3</td>
<td>1.6258</td>
<td>0.0384</td>
</tr>
<tr>
<td>H4</td>
<td>1.5872</td>
<td>0.0361</td>
</tr>
</tbody>
</table>
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

4.1.2 Trace Width

To analyze trace width variations, we re-used the aforementioned image processing approach where the 3 filter structures were imaged under a microscope. All of the images were aligned via Matlab image processing code, and then the raw files were converted to gray-scale. The gray-scale images were processed with a tuned ‘canny’ edge detection filter that produced a low noise delineation of the copper traces. The widths of the traces were measured by counting the number of pixels between opposing edges. A zoomed in example output from the filtering is shown in Fig 4.3

![Figure 4.3: Canny Filter Output](image)

Here, the white lines designate the detected edges of the copper trace. After passing the entire image through the filter, we should have a total of 32 edges detected down the center of the image. Each 2 edges, represents a copper trace with a gap of the same width in between. The larger cap between each interdigital structure was discarded for the purpose of measuring only the width of each trace. The length of a pixel was determined using a microscope calibration slide with 0.01 mm line
resolution. The calibration slide was imaged with the same zoom and focus levels that were used to image the NotchPUF structures. The measurements were divided by the pixel widths of the calibration sources to derive an estimate of the pixel length. From these measurements, each pixel was determined to be 0.002217 mm (or 2.217 µm). To account for noise (e.g., dust on the PCB when pictured), exactly 32 edges need to be detected down the measurement region. If this condition isn’t satisfied, the measurement region shifts to the right pixel by pixel until exactly 32 edges are found. This measurement region is shown in Fig. 4.4.

These measurements were taken on all PCBs, from all four panels. To further reduce the amount of noise that could potentially appear in the images (like dust collecting on the PCB), all boards were first imaged once taken out of their packaging. The board thickness was measured second, and finally the SMA connectors were soldered to each board. The results from this analysis are shown in the fig: 4.5.

Recalling previously, the designed width of each trace is 0.2mm. These results
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

Figure 4.5: Multi-panel Trace Width Variations

show that each panel of boards have mean distribution that is still within the 20% manufacturing tolerance. Panel H2 appears to have the most amount of variation with a few edge cases that fall outside of the 20% range of 0.16mm to 0.24mm range. These panel variations are summarized in the following table 4.2. These results are very encouraging for the design of NotchPUF, and shows that the mean trace width can vary approx 9-18% between panels of PCBs.

4.1.3 Substrate Relative Dielectric Constant

To be successful and have meaningful results, the dielectric constant experiment PCBs need to come from the same panel as the NotchPUF PCB. Because of the chosen board manufacturer, this wasn’t a guarantee. After the boards were received,
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

<table>
<thead>
<tr>
<th>Panel</th>
<th>Avg Width (mm)</th>
<th>3σ (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>H1</td>
<td>0.1930</td>
<td>0.0303</td>
</tr>
<tr>
<td>H2</td>
<td>0.1929</td>
<td>0.0449</td>
</tr>
<tr>
<td>H3</td>
<td>0.2014</td>
<td>0.0227</td>
</tr>
<tr>
<td>H4</td>
<td>0.2020</td>
<td>0.0327</td>
</tr>
</tbody>
</table>

Table 4.2: PCB Panel Trace Width Results

It became apparent that only 1 of the 4 dielectric constant experiment PCBs came from the same panel as their corresponding NotchPUF PCBs. Because of this, the experimental data couldn’t be directly correlated to the panel of NotchPUF PCBs. These challenges are explained in more detail in section 4.3.

Measurements of these 60 boards was still performed and their results compared to PCB panel H1. These 3 extra panels of boards are labeled Hx, Hy and Hz do differentiate them from the others. Following the same procedure outlined in section 3.5.3, $\varepsilon_r$ was calculated and summarized as shown in table 4.3.

<table>
<thead>
<tr>
<th>f(GHz)</th>
<th>H1 $\varepsilon_r$</th>
<th>Hx $\varepsilon_r$</th>
<th>Hy $\varepsilon_r$</th>
<th>Hz $\varepsilon_r$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>4.86</td>
<td>4.84</td>
<td>4.84</td>
<td>4.85</td>
</tr>
<tr>
<td>1.5</td>
<td>5.02</td>
<td>4.98</td>
<td>4.99</td>
<td>5.01</td>
</tr>
<tr>
<td>2.0</td>
<td>4.99</td>
<td>4.97</td>
<td>4.99</td>
<td>5.00</td>
</tr>
<tr>
<td>2.5</td>
<td>4.93</td>
<td>4.95</td>
<td>4.96</td>
<td>4.97</td>
</tr>
<tr>
<td>3.0</td>
<td>4.91</td>
<td>4.92</td>
<td>4.91</td>
<td>4.90</td>
</tr>
<tr>
<td>3.5</td>
<td>4.85</td>
<td>4.86</td>
<td>4.84</td>
<td>4.84</td>
</tr>
<tr>
<td>4.0</td>
<td>4.98</td>
<td>4.96</td>
<td>4.95</td>
<td>4.95</td>
</tr>
</tbody>
</table>

Table 4.3: Multi-board Variation of $\varepsilon_r$ in FR-4 ($\varepsilon_r=4.8$) Substrate

While not directly comparable to their corresponding panelized NotchPUF PCBs, the results were as expected with a 3σ variation ranging from 0.22 to 0.35.
Recalling previously, we used an Agilent 8753E VNA and an 85047A S-Parameter test set to perform the measurements. This VNA is limited to a maximum of 1601 data points for any given center frequency and span. Because of this, three separate measurements were taken at frequency spans of 1GHz, with center frequencies of 1,2 and 3 GHz. These measurements were repeated to obtain 200 samples for each center frequency. These samples were then averaged by sets of 10, to give a total of 20 sampled measurements. These same measurements were taken on all 174 boards and distinguished by PCB panel number H1 to H4. Before measurements were taken, a full 2-port calibration was performed with the same method as described in the experimental setup section 3.6.1. After calibration, a data analysis is performed on the raw measured data from the VNA. Then, the modulus technique (section 3.6.2) is re-applied and the PUF ID results are investigated.

4.2.1 Raw Data Analysis

To help understand how the variation from each panel affects the response filter, we first consider the maximum insertion loss and center frequency for this analysis. Each 1, 2 and 3 GHz structures responses are analyzed separately to determine if the variations had a higher impact when comparing between structures. This data is presented in scatter plots with the different panels indicated with separate colors on the graph. Panels H1, H2, H3 and H4 are represented by colors red, blue, green and cyan respectively. All boards have 200 samples with every 10 points being averaged, for a total of 20 data points. All PCBs are then averaged together for each originating panel. This computed average is indicated in the following figures with a cross-hair having the same color as the originating panel. These averages are then summarized in the tables following each of the scatter plots.
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

The 1 GHz structure response is shown in Fig 4.6. Panel H1 appears to have a center frequency closest to the 1GHz target, differing by only approx 14 to 18 MHz, and -0.07 to -2dB when compared to the other boards.

![1GHz Structures Minimums](image)

Figure 4.6: 1GHz Multi-panel Structure Response

<table>
<thead>
<tr>
<th>Panel</th>
<th>Avg Insertion Loss</th>
<th>Avg Center f(GHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>H1</td>
<td>-28.6747</td>
<td>1.0345</td>
</tr>
<tr>
<td>H2</td>
<td>-31.2064</td>
<td>1.0484</td>
</tr>
<tr>
<td>H3</td>
<td>-28.5995</td>
<td>1.0520</td>
</tr>
<tr>
<td>H4</td>
<td>-30.6880</td>
<td>1.0520</td>
</tr>
</tbody>
</table>

Table 4.4: Multi-panel 1GHz Structure Response Average
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

The 2 GHz structure response is shown in Fig 4.7. Panel H1 appears to have a center frequency closest to the 2GHz target, differing by only approx 33 MHz and -0.61 to -3.28 dB when compared to the other boards.

![2GHz Structures Minimums](image)

**Figure 4.7: 2GHz Multi-panel Structure Response**

<table>
<thead>
<tr>
<th>Panel</th>
<th>Avg Insertion Loss</th>
<th>Avg Center f(GHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>H1</td>
<td>-36.7321</td>
<td>2.0715</td>
</tr>
<tr>
<td>H2</td>
<td>-44.3858</td>
<td>2.1037</td>
</tr>
<tr>
<td>H3</td>
<td>-36.1268</td>
<td>2.1083</td>
</tr>
<tr>
<td>H4</td>
<td>-40.0163</td>
<td>2.1054</td>
</tr>
</tbody>
</table>

**Table 4.5: Multi-panel 2GHz Structure Response Average**
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

The 3 GHz structure response is shown in Fig 4.8. Panel H1 appears to have a center frequency closest to the 3GHz target, differing by approx 43 MHz and -1 to -6.2dB when compared to the other boards.

![3GHz Structures Minimums](image)

**Figure 4.8: 3GHz Multi-panel Structure Response**

<table>
<thead>
<tr>
<th>Panel</th>
<th>Avg Insertion Loss</th>
<th>Avg Center f(GHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>H1</td>
<td>-40.6262</td>
<td>3.1335</td>
</tr>
<tr>
<td>H2</td>
<td>-34.4668</td>
<td>3.1771</td>
</tr>
<tr>
<td>H3</td>
<td>-41.8431</td>
<td>3.1751</td>
</tr>
<tr>
<td>H4</td>
<td>-39.3844</td>
<td>3.1722</td>
</tr>
</tbody>
</table>

**Table 4.6: Multi-panel 3GHz Structure Response Average**
4.2.2 Multi-panel NotchPUF Response Analysis

The bitstrings generated by the NotchPUF were analyzed on a per panel basis to identify 3 key things. First, if any panel variation contributed to significant changes to the uniqueness of the bitstrings being generated. Second, to identify which variation had a higher impact on the overall change in insertion loss and frequency. Finally, a statistical analysis of the previously used PUF metrics with the additional 128 PCBs.

Recalling previously, the NotchPUF response is reduced in amplitude over the original values by applying a modulus operation, which effectively wraps the values into a smaller region. An example of this using a modulus of 40 MHz and -10 dB is shown in Fig 4.9. Here, each color represents a different panel of PCBs and each cluster of points an individual PCB. This modulus operation accomplishes two goals. First, it translates the (x,y) points associated with each PCB into a fixed size region. In this example, the region is bounded by 0 to 40 MHz along the x-axis and -10 to 0 along the y-axis. The fixed size region in turn enables boundaries to be defined for the assignment of bit values, as shown by the four quadrants in the figure. Second, the modulus removes bias that occurs across PCBs but preserves the within-PCB variations, which improves randomness in the generated bitstrings.

From Fig. 4.9 there are cases in which the cluster of points for each PCB cross over a boundary. When this occurs during regeneration in the field, bit-flip errors will occur in the bitstrings. Although the bit-flip errors occur in our experiments because of measurement noise, we also expect this to occur if the temperature of the environment changes. In either case, we propose to use error correcting techniques to deal with bit-flip errors as others have done in previous work [70].

Applying this process to the data from the three filters on each PCB generates six bits of the PCBs ID. Three additional bits are obtained by processing the FBW...
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

Figure 4.9: Modulus of Raw PUF Response

data, and another six bits using the maximum rolloff variation on each side of the $S_{21}$ response. Therefore, a total of 15 bits can be generated for each PCB using the three filters. These results are summarized in the following Table 4.7.

<table>
<thead>
<tr>
<th>Panel</th>
<th>Mean Inter HD</th>
<th>Mean Intra HD</th>
</tr>
</thead>
<tbody>
<tr>
<td>H1</td>
<td>48.7%</td>
<td>2.8%</td>
</tr>
<tr>
<td>H2</td>
<td>45.8%</td>
<td>2.5%</td>
</tr>
<tr>
<td>H3</td>
<td>46.3%</td>
<td>2.4%</td>
</tr>
<tr>
<td>H4</td>
<td>44.7%</td>
<td>2.1%</td>
</tr>
</tbody>
</table>

Table 4.7: Multi-panel Mean Inter and Intra HD

Using these metrics to analyze all 174 boards and with the same modulus, we achieve an Inter-PCB HD of 45.1% as shown in Fig 4.10 is close to the ideal of 50%. The Intra-PCB HD of 2.2% sown in Fig 4.11. While close to the ideal value of 0%, it is well within the error correction capabilities of error correction methods. To increase statistical significance, the distributions were created using all 200 measurements
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

taken per NotchPUF PCB.

Figure 4.10: 174 PCB Inter-Distance

Figure 4.11: 174 PCB Intra-Distance

4.2.3 Summary

With all boards combined, the mean inter-HD and intra-HD results were very promising, especially for only a 15-bit generated bitstring. The 176 PCB inter-HD decreased by approx 3.6% and the intra-HD decreased by 0.6% from the 48.7% and 2.8% reported previously (section 3.6.3) with only 44 PCBs. The modulus operation performed exceptionally well again by uniformly distributing the points of data.

4.3 Challenges

Throughout the fabrication of the 176 PCBs from 4 separate panels, there arose some issues with the production of the boards and results that were generated. Higher priced manufacturing facilities tend to have a higher standard for Quality Control (QC) on their assembly line. For the purpose of these previous experiments, a lower cost solution was used to create a baseline of what can be expected when there is little to no QC present.
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

4.3.1 Requesting Combined Panels

For the purposes of these experiments, it was important that the NotchPUF and dielectric constant experiment (Section 3.5.3) PCBs come from the same Panel. While communicating this to the manufacturer, it wasn’t a 100% guarantee that it would happen since their panelization processes were almost completely automated. Because of this, only 1 of the 4 panels had both the NotchPUF and dielectric experiment PCBs. A solution to this would be to combine both designs into a single gerber file [72]. Slight modifications would need to be made such that each board is easy to separate from the adjacent combined PCB. After this change, the gerber file would then be uploaded, and let their tools figure out the optimized placement on a panel.

4.3.2 PCB Colors

Having different PCB colors made it easy to visually differentiate between the originating panels, but these colors added additional image processing difficulty with the trace width analysis. This is partially because each color responded differently to the microscope light, so each color had their own set of tuning parameters, such as the light intensity, white balance, saturation, and color temperature. These tuning parameters were used to create a definitive contrast change between a copper trace and the FR4 substrate which is necessary for the image processing algorithm. Green PCBs were determined to have the least amount of parameter changes from the microscope camera, and black having the most.

4.3.3 Defects

The lower priced board manufacturer that was used, produced a number of highly visible defects every panel of PCBs. A few of these can be attributed to the fact that
these \( \text{PCBs} \) are 2-layers. Since the copper layers are exposed (minus solder mask), that any mishandling of a board could potentially damage the top or bottom layers. The following is a non comprehensive list of defects that were found on some of the boards:

1. Combined traces
2. Missing traces
3. Trace pour
4. Factory damage
5. Bad panel

The first two items could be attributed to the fact that the NotchPUF trace widths were designed to be at the minimum that would be accepted from the manufacturer. In some instances, the copper traces in a interdigital structure were combined (Fig 4.12), or missing a trace (Fig 4.13) midway down the structure.

![Combined Trace Defect](image)

Figure 4.12: Combined Trace Defect
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

Figure 4.13: Missing Trace Defect

Copper trace pour was another item that wasn’t very consistent between PCBs. An extreme example is shown in the following Fig 4.14. Here, the 2GHz structures from two boards (from the same panel) are compared side by side (B34H1 vs. B36H1). B34H1_2G had an average trace thickness of 0.2127mm, and B36H1_2G had 0.1711mm. While these traces are still within the limits given by the manufacturer, the difference between the two is a little over .04mm.

Figure 4.14: Copper Trace Pour Defect
This difference appears to be a significant factor between the response of their 2 GHz structures. This is shown in Fig 4.15. Board B34H1 is circled in the upper left hand corner, and B36H1 near the bottom right. We can see there is approx -30dB and 120 MHz difference between the two.

![Figure 4.15: B34H1_2G vs B36H1_2G](image)

Factory damage can happen anywhere from being manufactured, getting packaged, or in transit for delivery. The majority of the identified unknown damage to these were primarily cosmetic (heavily scratched solder mask). In a few cases, it appears as if something sharp was dragged across all of the traces, damaging them before the solder mask was applied. Fig 4.16 shows an example of this from PCB
Finally, an entire bad panel of PCBs is a possibility. We had sent our the NotchPUF PCB design to a 2nd manufacturer and had no control over what else was going on the panel. Because of this, we received an entire batch of PCBs that had a copper plane around the perimeter of the board that was shorting the two input paths together. This is shown in Fig 4.17.
Chapter 4. Multi-panel NotchPUF PCB Variation Analysis

4.4 Conclusions

In this chapter, we took a closer look at the performance of the NotchPUF with an expanded dataset. We showed the Inter and Intra HD were approx 4.9% to 2.2% from their ideal values. These statistical results indicate that the NotchPUF is able to generate quality bitstrings. In addition to this, the observed variation between panels confirms sensitivity the NotchPUF has to tolerances within the same factory. With all PCBs considered, the modulus technique combined with PBW and roll off, proved very useful in generating bitstrings at only a length of 15bits.
Chapter 5

Future Work

The PCB implementation of the NotchPUF has been highly successful in demonstrating the concept of the PUF and investigating its performance. The PUF was shown to perform well against the metrics for PUF performance that are applicable for bitstrings of a small size. We also confirmed the variations that exists from within the same PCB manufacturer.

In this chapter, I present a discussion of additional work that will be of great value to strengthen the performance specifically of the NotchPUF.

5.1 Environmental Conditions Reliability

The most obvious extension of this work is to perform an analysis on the effects environmental conditions have on the performance and reliability of the NotchPUF. PUFs are judged based on a fairly fundamental set of criteria, with environmental results being one of them. Typically, a PUF is subjected to temperatures ranging from -40°C to 85°C (for industrial temperature range).
Chapter 5. Future Work

Currently, temperature simulations have been performed on the FR4 substrate with some high level results in section 3.6.3. While we have proven these simulations to be highly accurate, further work needs to be done to confirm these.

5.2 Bitstring Length

The minimum requirement for a bitstring length that will allow the NIST statistical tests to be run, is a length of 100 \[56\]. While a single NotchPUF was able to generate 15 bits, more are needed to be used for any cryptographic application. There are a few ways this can be accomplished. First, more NotchPUF structures could be placed around a PCB and multiplexed for measurement. This would allow the pre-existing structure to be used in its current form, thus eliminating the need for re-design and simulations. Another solution would be to revert back to a single interdigital resonator, and add more of these structures around the PCB. Post experimental simulations suggest that a single resonator structure would perform roughly the same, but with close to half the insertion loss currently present. A third method that could be used would be an expanded frequency range with a different substrate than FR-4. While we determined 3GHz to be an upper limit for the NotchPUF on FR-4, there exists substrates that have far better dielectric properties. The last solution that could increase the bitstring length would be to design new NotchPUF structures that are embedded into a multi-layer PCB. By embedding, the feature sizes decrease with relation to the distance of the ground plane.

Being able to perform a bitstring analysis with the NIST statistical suite will help to strengthen the NotchPUFs suitability for cryptographic applications and to better quantify its randomness properties.
Chapter 5. Future Work

5.3 Signature Generation

The current bitstring generation technique uses the 1, 2 and 3 GHz structures center rejection frequency, maximum insertion loss (lowest -dB), FRW, and points of maximum roll-off (slope of curve). While effective for generating bitstrings of length 15, there exists more entropy that can be extracted from the analog waveform. These waveforms would be an ideal candidate for current machine learning techniques. The training dataset could utilize the data currently on hand and be expanded with future data from environmental testing. Once trained, individual board signatures could be generated with additional error correction data already included.

5.4 Attack Threat Analysis and Mitigation

The inherent nature of PCB, when compared to their silicon counterparts, is that they are much larger and more easily modified. Because of this, the NotchPUF is in a good position to provide a solution. Analysis into the manipulation of this solution could provide a foundation for a new level of board authentication and trust. Other mitigations could be an alternative design to that of which was proposed in this dissertation. This area of research continues to have multiple areas of opportunity.

5.5 Detailed Defect Analysis

A detailed defect analysis once performed, could help quantify how the NotchPUF could be used to identify any malicious type behavior like trojan insertion, board modification, etc. By observing how the PUF response changes with a known stimuli, a PUF response analysis could potentially be able to identify the type of “defect” that has been inserted. This type of analysis could also prove useful by using the
Chapter 5. Future Work

results to confirm various simulation aspects of higher complexity, such as a single wire having a significant impedance changes as if it was being actively probed, etc.
Chapter 6

Conclusions

In this dissertation, details have been presented regarding the design and experimental testing of ArbPUF and NotchPUF. We have demonstrated that these two PUFs are practical and realizable. We have also demonstrated several variations of bit generation techniques and analyzed their strengths and weaknesses. The results of these analyses show that the bitstrings are unique, random, and reliable, all which are crucial requirements for the usage of either PUF. In addition to this, a more detailed analysis was done with a much greater sample size of PCBs for the NotchPUF. This helped to solidify both the design and purpose of the NotchPUF, which is to be able to generate a board level identifier to later be used for authentication.
Acronyms

**ARB** Arbitrator ................................................................. v

**COTS** commercial off-the-shelf ......................................... vi

**CPLD** Complex Programmable Logic Device ......................... 35

**CRP** Challenge-Response Pair ............................................ 3

**DAC** Digital-to-Analog Converter ........................................ 18

**EEPROM** electrically erasable programmable read-only memory .... 2

**EM** Electromagnetic ............................................................. 40

**FBW** Fractional Bandwidth .................................................. 42

**FF** flip-flop ........................................................................... 13

**FPGA** Field Programmable Gate Array .................................... 4

**HD** Hamming Distance .......................................................... 6

**IC** Integrated Circuit ............................................................. 3

**ID** identifier ................................................................. vi

**IL** Insertion Loss ................................................................... 43

**MUX** Multiplexer .................................................................. 10

**NDA** non-disclosure agreement .............................................. 65

**NIST** National Institute of Standards and Technology .............. 6
Chapter 6. Conclusions

PCB Printed Circuit Board ........................................ vi
PCB Printed Circuit Board ........................................ vi
PUF Physical Uncloneable Function ................................. v
QC Quality Control.................................................... 77
RF Radio Frequency .................................................. 36
RO ring-oscillator ..................................................... 2
SMA SubMiniature version A .......................................... 51
SRAM Static Random Access Memory .............................. 2
SV stress-induced voiding ............................................ 5
TCV Thermometer Code Voltages .................................. 16
TCVD Thermometer Code Voltage Difference ..................... 24
TDC time-to-digital converter ........................................ v
TP Tap Points .......................................................... 24
TRNG true random number generators .............................. 8
TV temperature-voltage .............................................. v
VNA Vector Network Analyzer ...................................... 50
References


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


