Biasing a ring-oscillator based true random number generator with an electro-magnetic fault injection using harmonic waves.

Jeroen Senden

Master Thesis

Committee:
dr. M.H. Everts
dr. A. Peter
dr. ir. F. de Beer

Institution:
University of Twente

Chair:
Distributed and Embedded Security

14-01-2015

Abstract

This thesis shows the effect of an electromagnetic fault injection on true random number generators based on ring oscillators. It tests several designs, including ring oscillators of equal length and unequal length. We found that the created designs with ring oscillators of unequal length are more prone to fault injection. This research also shows that injecting the frequency of the operating frequency of the ring oscillators results in high mutual information. Fault injection using an electro-magnetic harmonic signal has a global effect, but also has local effects. An injection close to a wire connected to the ring oscillators seems like a good injection area.

Contents

Chapter 1

Introduction

Cryptography has been around for ages and is the main reason why we can communicate safely in the digital world (for example in internet banking). Most cryptographic functions need a random number, which is unpredictable, in order to work. This random number is used for a lot of cryptographic functions, such as the creation of a secret key, an initialization vector to start of a cryptographic algorithm or to prevent replay attacks. Should this random number become biased, the whole cryptographic function would become insecure.

There has already been extensive research done that describes an attack on the random number generator (RNG) by using a laser or electromagnetic (EM) waves. Recently, EM fault injection (FI) by harmonic emission (HE) has become a hot topic since it is a new area of research and countermeasures are not implemented most of the time. This document will be the basis of a research to investigate whether ring oscillator (RO)-based true random number generators (TRNG) in high-end targets can be biased by EM FI using harmonic emission. Before going into details, a scenario that explains why random numbers are important and an introduction with basic information will follow next.

1.1 Scenario

This section will describe some possible scenarios that could occur when random numbers are not random anymore. Figure 1.1 shows a scenario in an authentication setting where a bad random number causes the protocol to be vulnerable to a replay attack. If the random number would not be biased, a replay attack would not be possible. The protocol is a public/private-key authentication protocol. A user authenticates himself by decrypting a message that only the user can decrypt by using his private key. In this example Bob authenticates himself to Alice. Eve can eavesdrop on their communication and wants to authenticate as Bob to Alice, which should not be possible if the protocol is safe. If the random number was truly random it would prohibit Eve from doing a replay attack.

Bob will make the initial communication to Alice that he wants to authenticate to her. Alice sends Bob a challenge, which is a random value (R) encrypted by the public key of Bob. Bob is the only one who can decrypt this correctly using his private key. He gets the R out of the decrypted message, encrypts it with the public key of Alice and sends it back to Alice. Alice is the only one who can decrypt Bob's message by using her private key. If the R sent by Bob is the same as the R send by herself initially, Bob is truly Bob.

The messages that Eve has is the initial communication of Bob to Alice, an encrypted packet containing R and another encrypted package containing R. Eve has no idea what R was in this communication, since both packets were encrypted and she doesn't have the necessary decryption keys. Eve starts her communication using the initial message sent by Bob. If the same R is created, Eve will receive a packet which is the same as the one Bob received. She then knows what she needs to send back (although the contents of the packets look like gibberish to her) and Alice will think that Eve is Bob, since the two random values R are the same. Note that this scenario will also work if R is based on a small subset of values. Eve only has to eavesdrop on multiple communications in order to make the chance large enough that the challenge she receives is in her subset of eavesdropped communication.

Although the previous scenario is just theoretical, bad random number values have occurred in practice in the past. The most famous case of a broken RNG is the Mifare Classic, a contactless smart card. Nohl et al. [25] showed that they could consistently create the same nonce (number used once), computed with the same initial value on a Linear Feedback Shift Register (LFSR). The randomness would come from timing, and it was used for authentication. If one knows the nonce, only two messages are necessary to retrieve the secret key from the card with the help of precomputed rainbow tables. In this case not only the RNG was predictable when you controlled the timing, but also a bad initial seed played a vital part in the success of this attack.

Another example of a bad RNG was the SecureRandom java class on Android, which sometimes produced the same random value. This function was used by several applications, including bitcoin wallets. A bitcoin wallet is a wallet that stores your amount of bitcoins, a digital amount of money. A private key, a certain 'address' of the wallet and a random number are used for signing transactions. Due to the nature of the signature scheme, the private key can be discovered if it is used in two transactions with the same random value and same address. The bitcoin wallets used a deterministic RNG. Bitcoin transactions are also publicly available, which makes it easier to find vulnerable transactions. Private keys were thus leaked and malicious transactions were performed. Shortly after it got fixed, the same vulnerability was found for the JavaScript version, which again resulted into malicious transactions. This shows that a bad RNG can cause serious damage.

These are just some of the possible scenarios that have happened. It shows that RNGs need to be good and are a vital part of a cryptographic system. Bad RNGs could cause a complete cryptographic system to be undermined and render it useless and can cause serious damage. The next section will provide

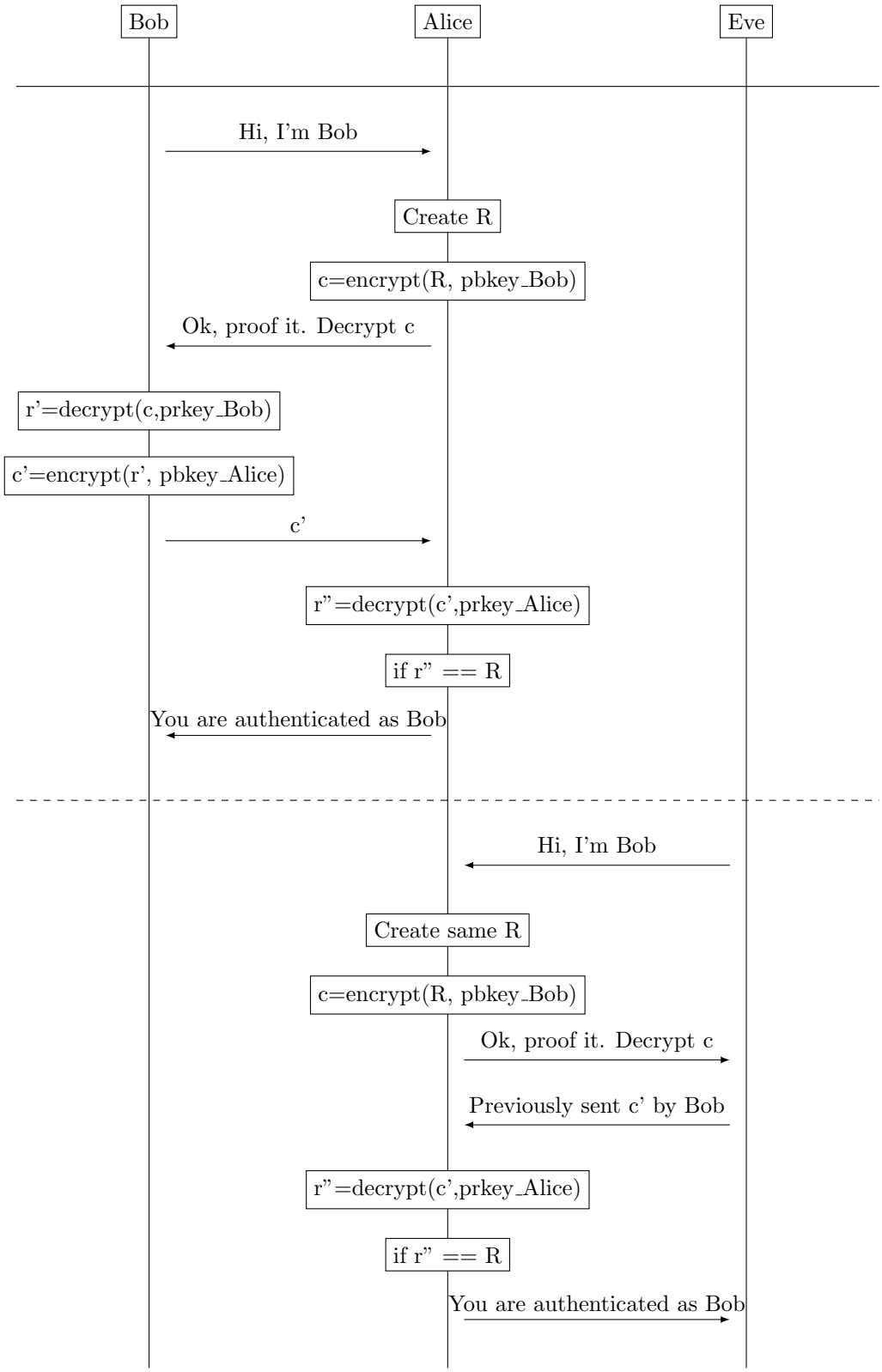


Figure 1.1: A authentication protocol using the same random value

some basic information on RNGs, followed by a quick look into some possible attack methods.

1.2 Random number generation

There are two types of RNGs. First there is the pseudorandom number generator (PRNG). This random number generator does *not* generate truly random numbers, but generates statistical random numbers. A number is statistically random when it contains no recognizable patterns or regularities and is calculated in a deterministic system. Although these pseudorandom numbers are not truly random, they are important nonetheless. The generation speed is fast and reproducibility is easy in most cases. The second type of RNG is the true random number generator (TRNG). This random number generator does generate truly random numbers and cannot be predicted since they do not rely on previous outcomes. For cryptographic functions a TRNG is preferred over a PRNG. This research focuses on TRNGs and PRNGs are out of scope.

In order to create a random number, one needs an entropy source, a mechanism to harvest this source and sometimes post-processing:

- The *entropy source* is the most crucial, since this will determine the randomness. The entropy source for a TRNG is a random physical phenomenon. A PRNG can collect a number from a true random number generator and run a deterministic function on top of it to create pseudorandom numbers. For example, some operating systems use disk input/output as an entropy source.

- In order to 'collect' entropy, a *harvesting mechanism* is needed. Some RNGs employ a XOR as a harvesting mechanism. A XOR take 2 bitwise inputs. If they are both the same, the output is '0', otherwise the output is '1'. If one of the inputs of the XOR is random, the outcome will also be random, making this an excellent harvesting mechanism for a RNG.

- A *post-processing phase* could be added to strengthen the RNG. The advantage of a post-processor is the fact that it could compensate for environmental changes or tampering. The disadvantage of a post-processor is that it will most likely degrade the output speed of random numbers. A common post-processor utilizes the von Neumann algorithm. The truth table for this algorithm is shown in Table 1.1, where x and y are 2 bitwise inputs to the algorithm.

There are some important features a RNG needs to have to prevent predictability. One of these features is that it needs to produce different random numbers each time it is restarted with the same initial value. Later on some TRNGs that need some initial time in order for them to generate random numbers will be shown (Chapter 1.4).

Although a RNG might produce random numbers at first sight, they might not be random nevertheless. There are several suites available in order to verify

x	y	out
0	0	-
0	1	0
1	0	1
1	1	-

Table 1.1: Truth table for the Von Neumann post-processing phase

if the generator creates (statistical) random numbers. The DieHarder test-suite[8] and the NIST SP 800-22 test-suite[26] are commonly used, since they test the most statistical properties that could exist in the random numbers and would thus not be statistically random. Note that these tests cannot guarantee that a RNG only produces random numbers. It can only prove that RNGs produce biased random numbers and are bad RNGs.

1.3 Attacks

To understand the attack that this research proposes, one needs to understand the different methods of attacking a target. A target is the system under attack, which can be any device that is a security critical system. Attacks can be categorized into groups. The first criteria is based on whether the attack is active or passive and the second criterion is based on whether the attack is invasive, semi-invasive or non-invasive. Note that an attack categorized by one criteria can also be categorized in the second criteria. The two different criteria are described below.

1.3.1 Active vs. Passive

When an attack is active, this means that the attack entails tampering with the target. This tampering can cause unforeseen or abnormal behavior, resulting in for example revealing the secret key. A passive attack is the opposite of an active attack. A passive attack monitors the target (e.g., power consumption and execution time) to determine for example a secret key. In an active attack, the target is thus manipulated to do some unforeseen behavior, whereas in a passive attack the target is executing according to its specification.

1.3.2 Invasive, semi-invasive, non-invasive

An invasive attack normally depackages the target and directly accesses parts of the target. Depacking the target makes it possible for the attacker to extract memory. In a non-invasive attack, the target does not get depackaged. An example of this is power analysis (monitoring the power that is consumed) of the target. A semi-invasive attack sits in between these two attacks. A semi-invasive attack does depackage the target (e.g., remove the silicon layer from a

smart card), but does not directly interact with the target (shooting a laser at the depackaged target does not directly interact with the target).

1.3.3 Fault injection

A fault injection (FI), which is the focus of this research, is always an active attack and can be either invasive (for example the previously mentioned voltage glitching attack) or non-invasive (for example by shooting a laser). A fault injection can be done in several ways. Lasers could be used to trigger some effect, the supply voltage of the chip could be altered shortly or an electromagnetic wave could be send towards the target. The idea of a fault injection is to make the target execute unwanted behavior, e.g. skip a line of code (software) or create a fault in the memory (hardware). When doing a FI, there is the risk of making the target incapable of resuming its normal functionality.

1.3.4 This research

This research will employ a FI using EM harmonic waves. The attack of this research will be active and non-invasive. It will be an active attack because it is trying to bias the TRNG, but non-invasive since it is trying to approach the high-end target intact and contactless. Because we want to do it non-invasive, no evidence of an attack is left on the target. It is targeted towards the hardware implementation of a TRNG.

Since TRNGs in high-end systems can employ TRNGs using ROs, it is of importance that these TRNGs are safe and do not become biased. Research has to be done to determine possible vulnerabilities, such that countermeasures can be placed where necessary. The amount of research done on possible vulnerabilities for TRNGs using ROs is very small. The research that has been done shows that ROs are vulnerable, but this research focuses on a specific small subset of ROs and leaves open questions. Further research needs to be done in order to verify that TRNGs using ROs are safe or whether these TRNGs need to employ counter measurements.

This document will continue with a quick look into the several random number generators that exist up till this date of writing (Section 1.4). Chapter 2 will contain relevant current research on EM-FI attacks and attacks focused on random number generators. This is followed by a chapter in which a research question will be formulated.

1.4 True random number generators

There are several ways to implement a TRNG. The next section will discuss the basics of a TRNG, followed by an overview of a TRNG using ROs. This research focuses on the generators based on ring oscillators (ROs), but Appendix A gives some insight in other entropy sources of TRNGs can be useful to grasp the inner workings of the reason a TRNG creates truly random numbers.

1.5 Noise

All RNGs need some kind of entropy on which the randomness is based. This is also called noise. Most RNGs are based on two types of noise: shot noise and thermal noise. Both will be discussed below.

1.5.1 Shot noise

Shot noise can occur in two systems, electronic devices and as optics. Electronic noise was first introduced by Schottky [29] in 1918. He studied the fluctuations in vacuum tubes. This kind of shot noise is based on the fluctuation of the electric current. This electric current has a certain amount of particles, called electrons, which are independent of each other. Optic shot noise relates to the counting of photons. Just as in an electric current, light consists of particles, in this case photons, which are independent of each other. Measuring the fluctuation in light is random and can be a quantum process.

1.5.2 Thermal noise

Thermal noise, also known as Johnson–Nyquist noise, is noise generated by the thermal agitation of the charge carriers inside a conductor. It can, for example, be used to let an inverter make a choice. An inverter is a element that converts a '1' into a '0' and vice versa, meaning that a stable state of an inverter always has a different output than its input. Consider an inverter has an input of '1' and an output of '1', which can be made possible by using transistors. Turning these transistors off (i.e., resistance set to zero), resulting in no control of the amount of electrons flowing through the conductor anymore, makes the inverter then decide whether the output or the input should become '0', because an inverter wants a different input with respect to its output. In a perfect world, the inverter would not be able to choose, but in the real world a small random atomic vibration caused by thermal noise makes the inverter go to either state. This principle was used by Intel in their random number generator presented in 2011[36].

Earlier, thermal noise was used by Holman [18] to create a high performance, continuous, non-deterministic RNG. The RNG is implemented on a CMOS, but could be applied to any integrated circuit (IC), as long as it consists of a low noise bipolar transistor. Xu et al. [41] implemented a thermal noise TRNG by only using 20 transistors and injecting it with a hot-electron.

1.6 TRNG using ring oscillators

This research focuses on TRNG based on ring oscillators. There are two different types of TRNGs based on ROs. First an overview of the basic working of RO-based TRNGs is given, followed by the two different types of operation that RO-based TRNG can have.

1.6.1 Theoretical overview

A RO consists of multiple inverters chained sequentially. The number of inverters chained is uneven and the last inverter is input for the first inverter, thus making it a ring. The last inverter is the input to the harvesting mechanism. Since the amount of inverters is uneven, the input of the harvesting mechanism keeps alternating between '0' and '1'. This is also depicted in Figure 1.2. As explained, an entropy source is needed in order to obtain a TRNG. In a RO-based TRNG this entropy is the jitter which is caused by the timing of the output signal (the input signal to the harvesting mechanism). This output signal is not a perfect square wave form (see Figure 1.3), which makes it unpredictable at what time the transition from '0' to '1' or vice versa takes place. This is also depicted in Figure 1.4. The RO is not a perfect square wave form because of e.g. temperature influences. For example, if the temperature is above a certain value, the propagation delay of the signal will be slightly higher and the operating frequency of the RO, the rate at which the RO is oscillating, will be slightly lower (and vice versa when the temperature is under a certain value). Jitter can thus be seen as the variation of the RO period. In Figure 1.4 the multiple rising and falling edges visualize this variation of the RO period. The jitter is based on the number of inverters multiplied by the delay of an inverter. In a RO-based design, there are usually more rings, although a RO-based TRNG can consist of only 1 ring. Multiple rings are used to achieve a higher output rate, but the jitter is also less susceptible to bias with multiple rings. There are ROs that employ a phase-locked loop. A phase-locked loop doesn't have the growing variation of the RO period as shown before in Figure 1.4. It has a given period (still dependent on random phenomena like temperature), but the sampling speed of the RNG is chosen on a frequency such that it samples exactly on the transition from high-to-low or vice versa. Figure 1.5 shows this, where f_{RO} is the frequency of the RO, f_{CLK} is the sampling frequency of the RNG and out is the output of the RO. Note that Figure 1.5 only shows a trace of a single RO and that a TRNG can employ more ROs.

The output of an RO-based TRNG is fed into a harvesting mechanism. The data output of the different ROs can be combined using different techniques. One technique is to use coupled oscillators, while another technique is to XOR the output of all the rings. There are more harvesting mechanisms that have been reported, but XORing the output of the ROs is the most common. When using more than 2 ROs, a XOR-tree is used. In a XOR-tree, the first XOR has 2 ROs as input and outputs the result to the second XOR. The second XOR takes this output from the first XOR and the third RO as input, and outputs the result. This is input for the third XOR together with the fourth RO etc.

An optional last phase can be a post-processing phase. A post-processing phase like the von Neumann algorithm (see Table 1.1) would remove bias from the RNG, but would cause a lower output bitrate.

In order for a TRNG to function properly, it needs high entropy. In order to have a high entropy, the source of randomness needs to be as independent from other characteristics as possible. Kyung Yoo et al. [42] investigated whether a

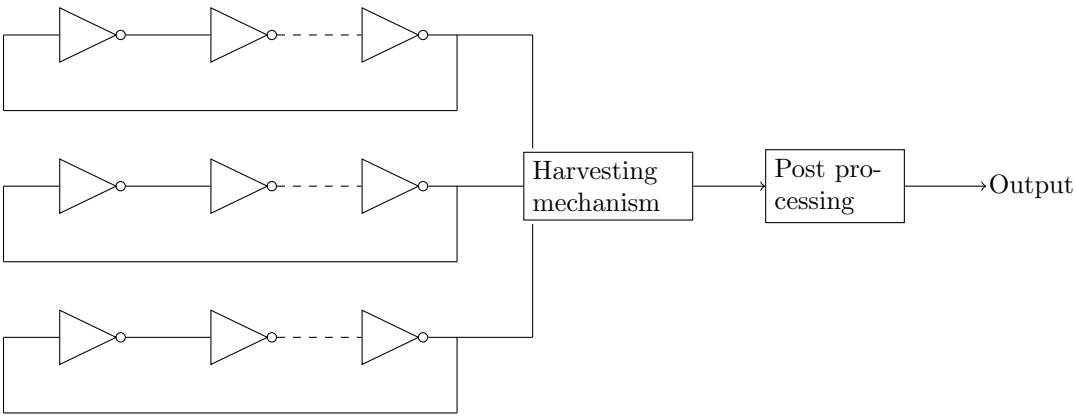


Figure 1.2: RO architecture

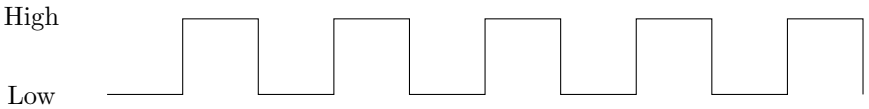


Figure 1.3: Perfect square wave form

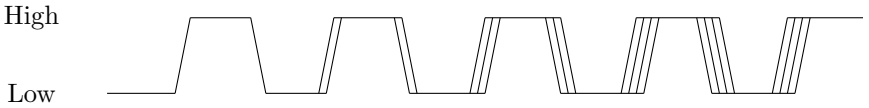


Figure 1.4: Waveform in practice

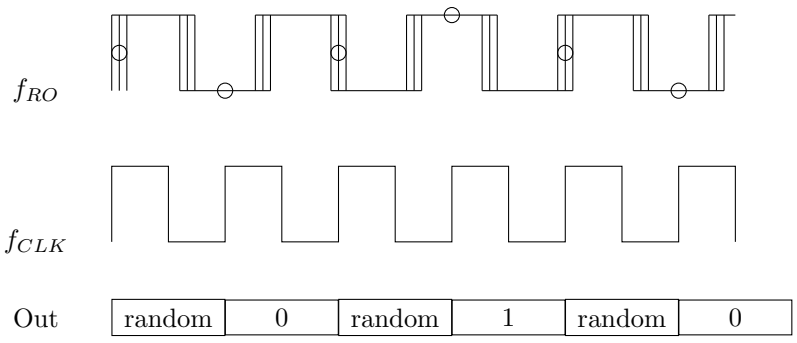


Figure 1.5: A phase-locked loop

RO-based TRNG is dependent on the supply voltage and the temperature. They show that it is susceptible to variations in supply voltage and temperature and that the sampling frequency could become a multiple of the oscillator frequency. This could mean that if ring r_1 transitions, ring r_2 could transition at the same time, resulting in wasted jitter since only one jitter is measured when XORing both rings. They propose an enhancement to the design in order to counter this effect. They propose to use rings of different lengths, such that it becomes less likely (for multiple rings) to shift all oscillation frequencies simultaneously to multiples of the sampling frequency. There are therefore currently two modes of operation, the first being a RO-based TRNG with rings of equal length, the second being a RO-based TRNG with rings of different lengths.

1.6.2 Equal ring length

When a RO-based TRNG has equal ring lengths, the number of inverters of every ring is the same. To decrease the chance of two rings transitioning at the same time and thus wasting jitter, more rings can be used. Sunar et al [35] discuss this concept. They propose to use a resilient function (post-processing) in order to keep the number of rings to a minimum. As discussed, the disadvantage of using a post-processor is the slow output of random numbers.

A follow-up on the research by Sunar was done by Wold and Tan [40], who show a system that does not need post-processing to pass the NIST and Diehard tests. The main difference between their system and the system proposed by Sunar et al. is an added D-flip-flop (which simply outputs the input (received at time t) at time t+1) after the ring and before passing the output to the XOR-tree. They elaborate on the fact that the bias in the system proposed by Sunar et al. comes after the XOR of the oscillator rings. The bias seems to be worse when more rings are used, causing a lot of transitions at the XOR-tree and sampling flip-flop.

A problem that might occur with RO of equal length, is that the ROs might synchronize with each other on a given frequency because their frequency might be closely related. A good example of this effect is the experiment with a lot of pendulum clocks that are out of sync, but eventually synchronize with each other after a while. This effect is also called mutual interlocking. Wold and Petrović [39] investigate the dependencies between the ROs themselves. It shows that interactions, correlations and dependencies exist between ROs that are implemented close to each other and operate on a closely related frequency. They also note that the amount of interaction, correlation and dependency is different between different architectures and thus different devices among different vendors.

1.6.3 Different ring length

In RO-based TRNG where the number of inverters are relatively prime to one-another, transitions are less likely to be occurring at the same time. This form of operandi should result in more useful jitter. However, Sunar et al. [35] give

a mathematical argument that using this form of operation is expensive due to choosing the correct sizes of the RO in order to retrieve an entropy that is good enough in order to pass the statistical tests.

Golić [14] introduced a RO based on a Fibonacci ring and a Galois ring. The combination of these two rings (first XORing the output signals before it is sent as input to a D-flipflop) is called a FIGARO ring. In a Fibonacci ring oscillator every output of the inverters is used as feedback for the first inverter. In a Galois ring oscillator every input to a inverter consists of the output of the first inverter and the output of the previous inverter. The advantage of combining these designs is the quick propagation of jitter and thus a quick, good entropy source. The mutual interlocking was also reduced and XORing it makes it more robust, resulting in a higher entropy. These rings are also easy to implement on a FPGA. A restart experiment was done to test the efficiency of the propagation of the jitter. Using the same conditions to restart a Fibonacci ring a 1000 times results in a standard deviation of almost zero in the beginning. After 30 ns the jitter propagated throughout the whole ring and the jitter becomes random. When doing the restart experiment with a RO of length 3, it takes much longer for the ring to have a random jitter (around 3000 ns). Using Fibonacci rings gives the opportunity to create good random numbers faster from a restart-state than using ROs of equal length. This is especially useful for smart cards, since smart cards lack a constant source of power.

This concludes the overview of RO-based TRNGs. Some additional TRNGs can be found in Appendix A.

1.7 Research questions

Research on FI using an EM-field in harmonic waves is still new. Many questions still remain unanswered and haven't been researched yet. This research will be a follow up on the research of Bayon et al. [6]. They implemented their attack on ROs of equal length. They also report that their ROs were located near each other. This has several advantages for their research. The first advantage is that the point of injection for influencing all the ROs is not an issue. Another advantage of using ROs of the same length is the fact that the operating frequencies (the frequencies at which the ROs oscillate) of the ROs will be close together. Besides these advantages, using ROs with only 3 inverters have a high frequency, which is beneficial for the speed of electric coupling and might influence ROs quicker [30]. Although this research was successful for their particular case, in reality a TRNG based on ROs might have a different design where it would not work. Furthermore, a logical countermeasure to this attack would be using ROs of different length, such that the frequency of the ROs is not so close together and thus an optimal injection frequency might be hard to find. It would even make sense to use ROs of different lengths in a high-end system that needs to be secure, since the frequency would differ and the space on the surface would differ. Indeed, if this causes ROs to be spread over the whole chip, this attack might become useless since the effect might be more

local in stead of a global effect. The main research question for this research will be:

- Is an EM-FI using harmonic emission attack on a different length RO-based design feasible?

This research can be extensive, since different length RO-based TRNGs will have different frequencies. Finding an optimum injection frequency can be hard to find in order to synchronize them in a way that all the ROs are not independent of each other anymore. The difference between the operating frequency of the ROs might become too large. Finding this threshold in difference can be useful (if it exists) since it could be a countermeasure for this attack. When using ROs of different length, the spatial aspect of the placement of the ROs on the chip can also become an issue. If ROs are not placed close to each other, this attack might become unfeasible. This research can also be seen as a stepping stone to see whether an attack using EM-FI using harmonic emission (HE) is feasible against a high-end target.

Although the area of research is still new, some successful attacks have already been reported. Some related work will be discussed in the next Chapter before going into the research done in this thesis.

Chapter 2

Related work

Electromagnetic analysis (EMA) on cryptographic systems has been extensively explored. However, EMA on TRNGs is fairly underdeveloped. An important reason for this is that cryptographic systems are larger and more complex and will hence give more electromagnetic emanation. In contrast, a TRNG is small and has a small electromagnetic emanation and is embedded in the cryptographic system most of the time, which makes locating and targeting of the TRNG hard. Finding the location of the TRNG is also called 'cartography'. This section will describe some of the research that has been done in cartography. Afterwards some of the attacks on TRNGs that have been researched will be given.

2.1 Cartography

In 2013, Bayon et al. ([4], [5]) described ways of determining the position and the operating frequency of a RO within a FPGA, while it is running an AES-algorithm. If such a location and operating frequency is known, an attack by Bayon et al. [6] becomes faster and easier. The frequency of a RO depends on the power supply and the temperature. If one could alter one of these dependencies, one can do a differential analysis to determine the location and frequency of the ROs. This is exactly what was done by Bayon et al., resulting in successfully locating the ROs whilst a cryptographic algorithm is running. They also showed that the sampling frequency can be easily obtained by obtaining a differential power spectral density for the whole circuit and determining the space between frequency peaks, which should be the same in the whole trace. Using this cartography technique reveals the location of the ROs and also the frequencies on which everything in the chip operates.

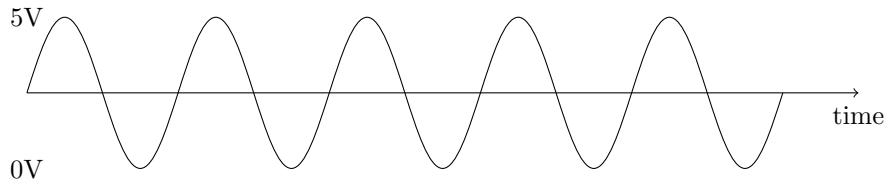


Figure 2.1: An EM harmonic emission

2.2 Attacks

Targeting the TRNG instead of the cryptographic system is a relatively new area of research. As explained in Section 1.6, a RO-based TRNG is influenced by the temperature it is operating in and the supply voltage. Simka [31] evaluated a RO-based TRNG on an FPGA with temperature fluctuations. He observed that it is still influenceable, but as long as the number of samples influenced by jitter is high enough, the TRNG is not biased and will still pass all the different statistical RNG tests.

Soucarros et al. [32] tested two different TRNGs operating at a different temperature. The first TRNG was based on thermal noise, the second RNG was an RO-based TRNG. The TRNG based on thermal noise got extremely biased without post-processing. When post-processing is applied, the bias can be removed. The RO-based TRNG did not get biased as much as the thermal noise TRNG, but a linear relationship is shown. The higher the operating temperature, the more bias occurs. Again, post-processing is able to remove the bias from the output. This research showed that TRNG are influenced by temperature and (in secure critical applications) a post-processor should be applied afterwards in order to unbias the output.

The research that triggered the EM research on RO-based TRNG was done by Markettos et al. [22]. Although they do not describe an EM attack, they do touch upon the subject of harmonizing the frequencies of the ROs, such that they transition at the same time, causing the jitter to be useless. If the jitter is useless, then the TRNG will output biased random numbers. Markettos et al. observed that they could phase lock the ROs to a certain frequency injected into the power supply. Markettos et al. build their research upon prior research done by Mesgarzadeh et al. [24] and Adler [1], who both showed the effects of an injection-locked RO (phase noise reduction and jitter reduction).

2.3 EM fault injection using harmonic emission

EM fault injection using harmonic emission continuously sends out a sinusoidal wave, as shown in Figure 2.1. The voltage, as shown on the y-axis, is dependent on the power of the injection. The x-axis shows the time. When injecting faults, one chooses a frequency and a injection power.

One of the first to start doing EM fault injection using harmonic emission

on an IC are Alaedine et. al [2]. They tested whether an IC is sensitive towards EM emissions. They show that an IC is not only sensitive to a magnetic field, but even more sensitive to an electrical field.

Poucheret et al. [27] applied EM harmonic emission to an integrated circuit running a RO-based TRNG. It describes how it affected the output frequency of the RO. This was mainly due to the power ground network, which made it possible for the injection probe to couple with the circuit. Poucheret et al. were able to increase the output frequency of the RO by 50%. This makes this a serious threat, because this gives a large window to lock the frequency to a multiple of the sampling frequency, rendering jitter useless.

In 2011 Hayashi et al. [17] showed an effective attack on a cryptographic system running an AES algorithm. By means of differential fault analysis (DFA) they were able to determine the key. The attack used a sinusoidal wave, but an injection probe was directly attached to a power line of the IC. The sinusoidal wave could be created from a 60cm distance to create effective faults, and no precise trigger was used to inject the fault. They touch upon the subject that this injection probe should not be necessary and that an antenna can also be used.

Bayon et al. [6] investigated the effect of EM-FI by harmonic emission on a TRNG based on ROs. In 2012 they showed that it was possible to completely bias the output of a 50 RO-based TRNG (the one proposed by Wold et al. [40]), up till a point where they could tell the TRNG what to output by dynamically adjusting the EM emissions. They could alter the RO output to produce only zeroes, indicating the ROs were all interlocked and thus outputted the same value. When the result of every RO is the same, the harvesting mechanism used (a XOR-tree) always outputs a '0'. They also showed that more injection power yields a better effect.

Buchovecká and Hlaváč[10] show an invasive and a non-invasive variant of a frequency injection attack in order to 'stabilize' a RC oscillator, which is an oscillator consisting of resistors and capacitors. Their RC oscillator outputs 8 random bits per second. For the invasive method, they use a crystal oscillator operating at 8 MHz. They show it is possible to influence (and thus reduce the randomness of) all the generated bits using their invasive method. The non-invasive method consisted of a function generator that had a sinusoidal signal of 8 MHz, which was broadcasted by an antenna. Although the non-invasive method does not influence all of the generated bits, bit numbers 6 and 7 (the two highest bits) were still significantly biased, resulting in significantly less unique values. This research shows that not only true random number generators based on ring oscillators are vulnerable to this kind of attack, but other true random number generators also. Further details can be found in [9].

Hadáček also did some experimentation on an RC oscillator, although the research does not go into the details. He showed that the RC oscillator started functioning slower. This did however not influence the quality of the generated random bits.

2.4 EM fault injection using pulses

EM fault injection using pulses is mostly targeting the cryptographic system. Dehbaoui et al. [12] show that the fault they injected using an EM pulse is data-independent on a cryptographic system running AES. This means that most DFA schemes are possible to implement. Schmidt et al. [28] managed to factorize a CRT-based RSA modulus by using a spark generator.

Velegati et al. [37] present a experimental setup and elaborate on the different aspects of the coil and its impact on a target. They also discuss the steps for calibrating and conducting an EM FI. They tried to fault a simple counter in an Android ARM core, but did not succeed. They did induce other faults into the ARM core, suggesting it is vulnerable to EM FI. Further research will need to be done (fine-tuning of parameters) to eventually fault the simple counter, after which a cryptographic algorithm can be targeted.

2.5 EM countermeasures

Zussa et al. [45] investigated whether voltage glitch detection mechanisms and clock glitch detection mechanisms can counter EM fault injection with pulses. Since EM introduces drops in the currents of the IC and changes the propagation of signals, these mechanisms could work. The only difference is the spatial effect of the EM fault injection in respect to voltage glitching, where EM fault injection can act locally and the voltage glitching is global. Therefore, more of these countermeasures were implemented in the IC, but still several faults were not detected. They do not elaborate on the effects of EM fault injection by harmonic emission.

Hayashi et al. [16] also touch upon the subject of revisiting ferrite cores as a countermeasure against EM fault injection in order to provide security to the legacy parts of the system that did not receive any security, since they show that EM fault injection can affect a cryptographic system through these legacy parts of the system.

Although not a countermeasure, Alberto et al. [3] investigate a way to determine the effects of an EM attack before it gets send to the manufacturer. Sign-off power analysis seems to be a good way to identify parts that are more error-prone to EM FI which need a higher margin of tolerance in power fluctuation. Voltage (IR) drop analysis can more precisely identify highly sensitive parts where knowing the acceptable margin of tolerance and observing the errors may allow evaluating the actual transferred power.

Chapter 3

Setup

This Chapter will give an overview of the setup used, as well as the different probes that were used and the different targets. It will also elaborate on the methods used to verify a good injection frequency, explain the calculation of the mutual information (MI) and give the RNG test-suites that were used.

3.1 Overview

This section describes the setup that was used. An overview can be found in Figure 3.1.

The signal of the last inverter element of the RO is routed to an output pin, to be able to measure the signal. An oscilloscope, a LeCroy, is used to measure this signal. The LeCroy transmits the data to the laptop were further analysis is done. Analysis was done on a laptop using Inspector, a software tool created by Riscure for side channel analysis and fault injection.

The laptop also controls the signal generator (the injection power and the injection frequency). It can be controlled using the software shipped with it, or using an external Python script. Inspector can call this Python script, making this a very flexible system. The amplifier is also located on an XY-station which can also be controlled by Inspector. The laptop is also connected to the target with a USB-cable. Getting a random number from the target can be done from the command-line using the provided program.

The laptop had a connection to a flash programmer that was connected to the FPGA's JTAG. The flash programmer was used to program the FPGA with a desired TRNG design.

The signal generator feeds a signal into the amplifier, which is hooked up to an external power supply. The amplifier transmits the signal to the probe, which is then partly forwarded into the target (and the open world) and partly reflected back. This setup has no means to measure the power transmitted by the probe, but only knows the input powers to the amplifier.

This setup has three differences compared to the setup used by Bayon et al.

The first difference is the amplifier. The second difference is the probe used to inject the signal. The length of the probes used for this research do not have the same length as the probe used by Bayon et al. The last difference is the measurement point to identify the power emitted by the probe. Bayon et al. were able to measure the output power of the probe, while this is not possible in the setup used in this research.

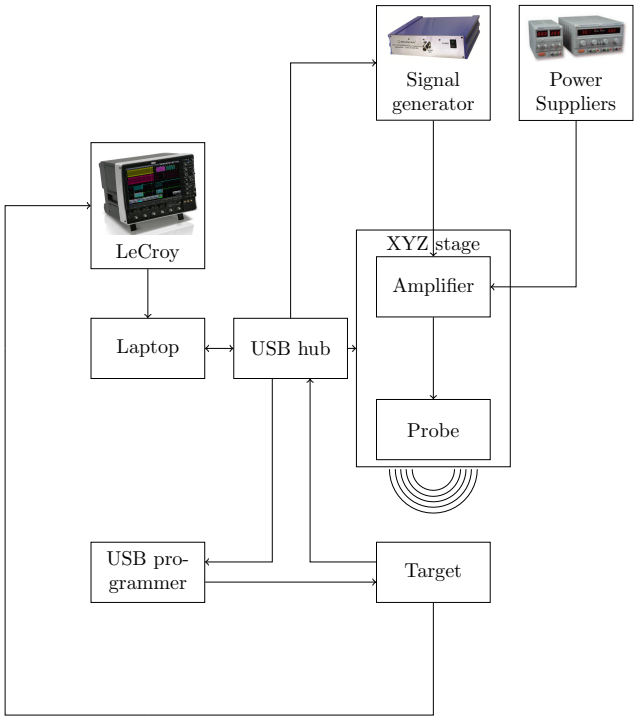


Figure 3.1: Overview of the setup used for this research

3.2 Probes

For this research two kinds of probes were used to inject the harmonic signal. They were both used to see the different kind of effects that a probe might have. The difference between these two probes is the length of the probe and the shielding of the probe. The short probe is approximately 5 mm long and the long probe is 51 mm long. Both probes have a diameter of 0.125 mm. Because the length of one probe is longer, it is assumed that the effect of the injection with that probe is is stronger compared to the shorter probe. It is also assumed that the longer probe has a larger area of effect. However, the short probe is also shielded and might therefore give a more localized effect than the long probe. If positioned on a good spot, the short probe is expected to influence only the ROs

and not the rest of the design running on the FPGA. The long probe should have a bigger effect, but is assumed to also influence the rest of the design on the FPGA. The experiments performed with the short probe can be found in Appendix D.

3.3 Targets

This thesis also investigated the effect on two different targets. The first target is named TestTool and is developed by Riscure and is used as an internal evaluation board. TestTool has a Xilinx Spartan-6 FPGA. The second target is the same FPGA that was used in the research performed by Bayon et al., which is an Actel Fusion M7AFS600.

The Actel Fusion FPGA could be programmed using a flash programmer. A design for the TRNG can be created in the software named 'Libero', shipped by Microsemi. The design used by Bayon et al. was used as the base for the created designs for this research. TestTool could be programmed using software called 'Vivado', shipped by Xilinx. In contrast to the Actel Fusion, TestTool does not require a flash programmer in between, but is connected to the laptop with a USB Standard B–plug. The main focus of this thesis will be on the Actel Fusion FPGA. The research performed on TestTool can be found in Appendix B.

3.4 Verification methods

This section describes two methods to find the optimal injection frequency and a method that determines whether the chosen injection frequency is also performing as expected. The first method for finding the optimal frequency is adopted from the paper by Bayon et al [6], while the second method is derived from results from this research. Both methods will be explained in Section 3.4.1. In order to see that our injection is locking the ROs, mutual information was used and explained in Section 3.4.2. RNG test-suites were used to check if the TRNG was biased. These can be found in Section 3.4.3.

3.4.1 Finding the optimal injection frequency

The general method for finding an optimal frequency starts with performing a frequency sweep. The optimal injection frequency can be lower or higher than the operating frequencies of the ROs, but also a frequency in between the operating frequency of the ROs. Bayon et al. did a frequency sweep in a lower range of frequencies than the operating frequency of the ROs. This does not mean that an optimum injection frequency can not be higher or equal to the frequency of the ROs. The average frequency of all the operating frequency of the ROs should be the optimal injection frequency from a logical point of view. Unfortunately working with these high frequencies does not always have foreseeable consequences.

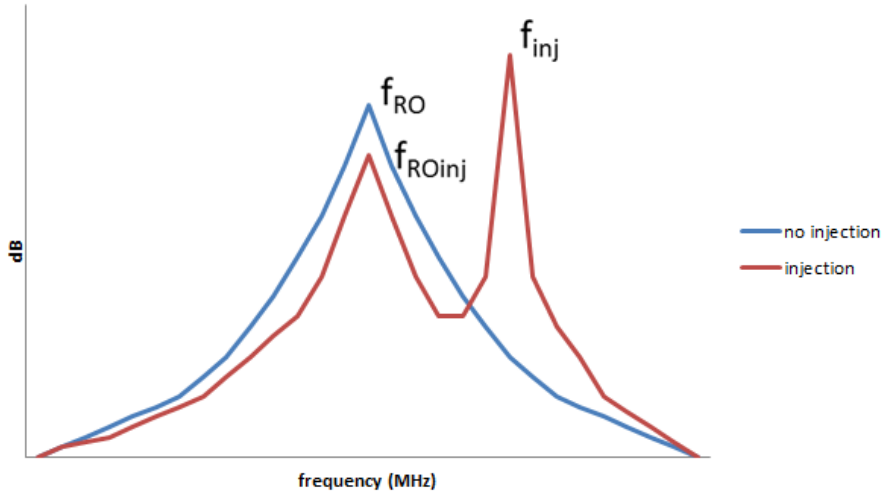


Figure 3.2: Simplified example, showing 2 traces (with and without injection) in the FFT-spectrum.

When finding the optimal injection frequency, there needs to be some kind of method to measure the operating frequency of the RO. In our case, the output signal of the last inverter of the ring was routed to an output pin. Another possibility is to measure the EM-signal emitted from a certain area of the chip. If both ROs are closely together, a clean signal of only one RO can be hard to get. The measured signal is a waveform, where the x-axis is the time and the y-axis will be the voltage.

Two methods to find the optimal injection frequency are described below. Both methods require a trace from the signal of one RO. A Fourier transformation is applied to this trace, which shifts the trace into the frequency spectrum. In the frequency spectrum the frequency of the RO will be visible and (if the injection power is strong enough) the injection frequency. Figure 3.2 shows a simplified version of a spectrum trace. The first method compares the RO peak (dB(f_{RO}))to the injection peak (dB(f_{inj})) and is the method used by Bayon et al. The second method is a method derived during this research.

Method 1: RO peak divided by the injection peak

In the frequency spectrum there are (at least) two peaks, namely at the operating frequency of the RO and at the injection frequency. There might be more (lower intensity) peaks visible in the spectrum, which can relate to the operating frequency of another RO or the frequency of an internal clock signal. Dividing the intensity of the injection frequency (dB(f_{inj})) by the intensity of the operating frequency of the RO (dB(f_{ROinj})) gives a certain value. The higher this value is, the more effective the injection frequency is. A high peak in this

spectrum signifies more activity on a given frequency. If the injection frequency peak is higher than the operating peak of the RO (which is the measured signal), the RO might have locked to this injection frequency.

Method 2: RO peak during injection substracted from the RO peak without injection

This method requires two cases. One case is a measurement without injection. The second case should be taken during injection. As mentioned before, a high peak in the frequency spectrum signifies a high activity on that frequency. To see whether ROs might have been locked to a frequency different from the original frequency, one can also measure the y-value of a peak in the spectrum at two different points in time. This method only looks at the height of the peak of the operating frequency of the RO. If the peak of the operating frequency of the RO during injection (dB(f_{ROinj})) is lower than the peak when no injection is done (dB(f_{RO})), it can be concluded that the RO has less activity on that frequency and locked to another frequency. An optimal injection frequency would then be the lowest value. Although this method is not described in the current literature, Section 5.2 shows that it yields similar results.

3.4.2 Mutual information

While the above methods aim to find an optimal injection frequency, this does not mean that an attack on the found injection frequency works. In order to verify that the injection frequency locked the ROs another measure is used: mutual information. Mutual information calculates the information in bits that is shared among two different entities, in this case ROs. When the optimal injection frequency is found, it can be verified using mutual information. Mutual information needs measurements of two ROs. These measurements are the voltage usage of the element of the ring that is connected to the harvesting mechanism. It gets the voltage level of these two measurements at a given sampling speed (10 GHz for example) and divides these points into a certain amount of bins. The mutual information is calculated from these bins. If the mutual information is (close to) zero, the two ROs are independent from each other. Mutual information is upper bounded by the minimum entropy of the amount of bins.

This research divided the different sampling points into four equally sized bins. For every trace the maximum and the minimum voltage level was acquired. The minimum was substracted from the maximum and divided by the number of bins (4 bins in this research). This gives the size for every bin. The first bin would thus be in the range $[minimum, minimum + 1 * size]$, bin two would be in the range $[minimum, minimum + 2 * size]$ etcetera. Once all the traces are processed and the points from the traces are divided into the bins, the mutual information is calculated. In this research the mutual information is upper-bounded by 2. If the mutual information is 2, the ROs are completely

interdependent and thus locked onto the same frequency. This means that the output of the TRNG should be completely biased.

3.4.3 Random number test suites

A definite way to check if the attack succeeded is to check the random number produced by the system. The NIST monobit test and block frequency test were used to check if the attack succeeded. The reasoning is that if two rings have a high mutual information, the resulting XOR-tree will produce a lot of zeroes. These zeroes are sampled, gathered into a binary file and fed as input to the test-suite. A test-suite should be able to determine if the fault injection was successful at biasing the TRNG based on the monobit-test. In order to account for temporary effects, a block frequency test was also used. If there are certain blocks that contain a lot of zeroes (or ones), this test should be able to find it. Other random number test suites like Dieharder and AIS-31 were also used. The advantage of the NIST test-suite is the low amount of bits required to run the tests, thus having fast results.

Chapter 4

Initial experiments on a TRNG

This chapter describes an initial experiment to monitor the effects of an EM-FI using harmonic emission on the RO-based TRNG running on the Actel Fusion FPGA. The operating frequency of an RO is primarily determined by the elements of the ring and the wires connecting it. Fluctuations on this operating frequency can be induced by the temperature and the injected frequency. During a FI the temperature of the FPGA rises. Reasons for this are the heat of the amplifier that is blown on top of the FPGA, but also the electric coupling in the FPGA induced by the injected signal. This experiment aims to identify the effect of the FI on the operating frequencies of the ROs. First some architectural decision that were made will be elaborated, followed by the experiments and results. A conclusion will summarize the results for this experiment.

4.1 Design

In order to have more effect on the ROs, an antenna was introduced in every RO. The distance between the first and second element of a RO in the TRNG was made larger. Due to this distance, a long wire connected these elements. It is assumed that a RO with this long wire is influenced easier than a RO without a long wire since the area of impact is larger. Figure 4.1 shows an example of a RO with and an RO without an antenna. A disadvantage of this antenna is the drop of operating frequency it causes. A lower frequency means less effect on the RO because the electric coupling behaves less effective.

For the next experiments, the ROs consisted of 3 elements. Without an antenna, the operating frequency of the RO would be in the window $[320 - 330]$ MHz. With the antenna, the operating frequency drops to $[240 - 260]$ MHz. This large window is based on the routing specifications of the FPGA that it implements and the optimum positioning of all the elements of the design. Once a RO is placed and routed, the operating frequency of the RO can change with

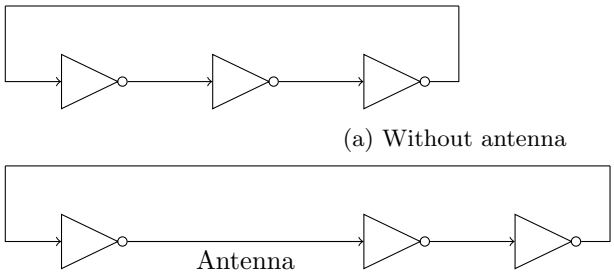
roughly 2 MHz (depending on temperature, injected frequency etc).

4.2 Initial experiments

To see what happens with the operating frequency of the ROs when fault injection takes place, an experiment was performed that monitors the operating frequency of the ROs in different points of time. This was done at two different injection frequencies and 3 different injection input powers. There are 2 ROs implemented in the FPGA, which have an operating frequency of roughly 260 MHz and 252 MHz. The two different injection frequencies are 220 MHz and 300 MHz and the injection input powers were set to -4, -2 and 0 dBm. Injecting on 220 MHz and 300 MHz was not chosen for a particular reason, except for the fact that both numbers are roughly 40 MHz lower and higher than the operating frequency of the first RO. Since temperature influences the operating frequency of the RO, the FPGA had no power for an hour. Although the temperature inside the FPGA could not be measured, it is assumed that this would lead to approximately the same temperature at the start of every experiment.

4.2.1 Injection of 220 MHz

Figure 4.2 shows the effect on the operating frequency of the ROs while injecting a harmonic signal at 220 MHz on different input powers. The first RO starts at a frequency of roughly 261.5 MHz for all the three different input powers. The operating frequency stabilizes after 50 minutes of injection. The second RO starts at a frequency of roughly 253 MHz and has a stable frequency after 30 minutes. For an input power of -4 dBm, the operating frequency of the first RO stabilizes at 258.33 MHz. For an input power of -2 dBm and 0 dBm the operating frequency of the first RO stabilizes on 257.87 MHz and 257.42 MHz respectively. The operating frequency of the second RO stabilizes on a frequency of 250.09, 249.63 and 249.33 MHz for an input power of respectively -4, -2 and 0 dBm. The higher the input power, the lower the operating frequency of the

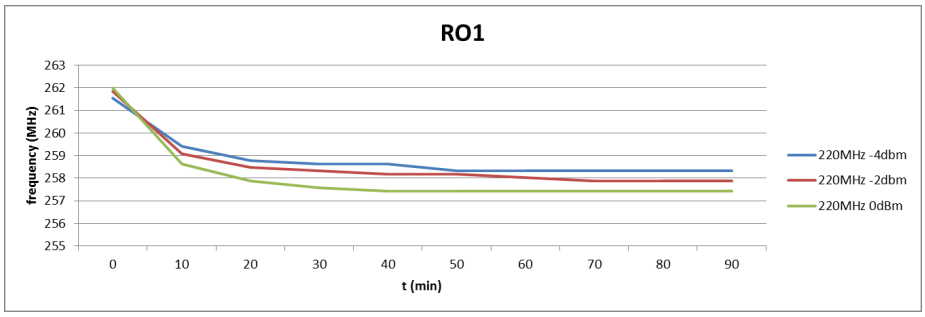


(a) Without antenna

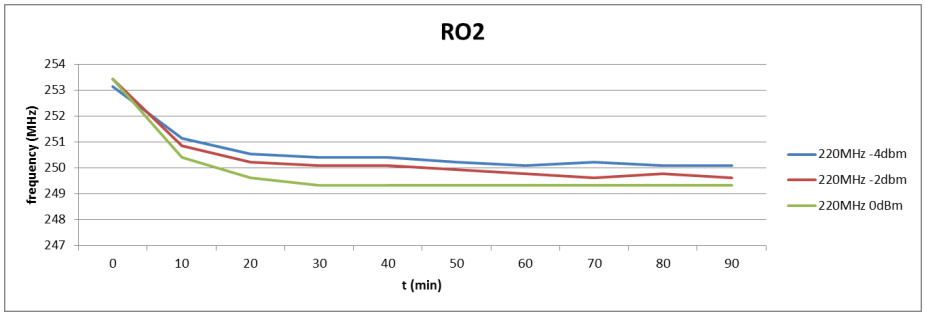
(b) With antenna

Figure 4.1: RO of 3 inverters

ROs becomes over time.



(a) Injection effects on operating frequency of RO 1

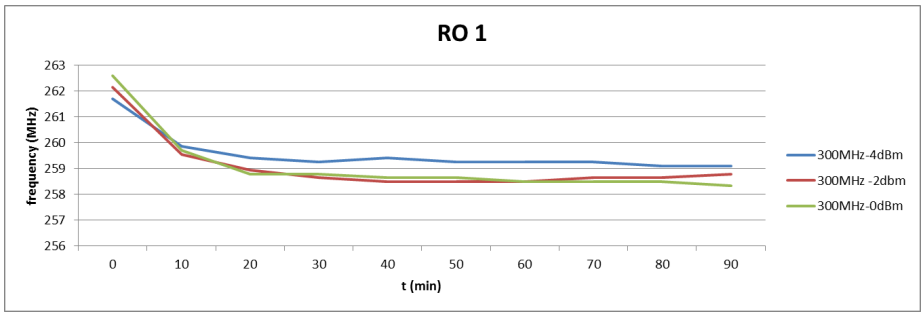


(b) Injection effects on operating frequency of RO 2

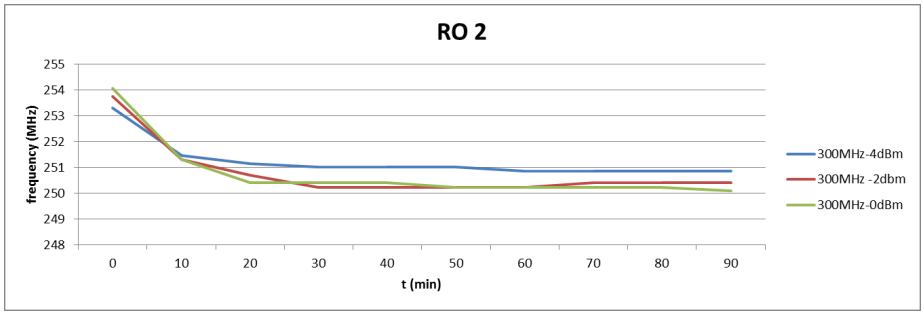
Figure 4.2: 220 MHz injection on 3 different input powers

4.2.2 Injection of 300 MHz

In Figure 4.3 the same behavior can be seen as when injecting with 220 MHz. The first RO starts at roughly 261.5 MHz and the second RO starts at roughly 253.5 MHz. After 50 minutes of injection the operating frequency of both ROs seem to stabilize. However, when injecting on 300 MHz the operating frequencies of the ROs stabilize to a higher frequency compared to injecting on 220 MHz. Injecting 300 MHz on 0 dBm results in roughly the same operating frequency of both ROs compared to injection of 220 MHz on -4 dBm. This might be caused by the fact that the injection is of a higher frequency than the operating frequency of the RO, which results in a slight pull up in the frequency. Another reason might be that the temperature induced into the chip by the injection is less at 300 MHz than injection of 220 MHz, depending on the power transmitted by the probe. Unfortunately, this power could not be measured in the current setup.



(a) Injection effects on operating frequency of RO 1



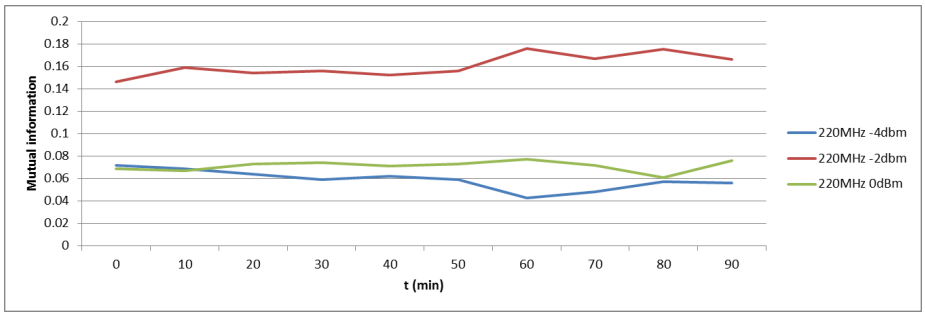
(b) Injection effects on operating frequency of RO 2

Figure 4.3: 300 MHz injection on 3 different input powers

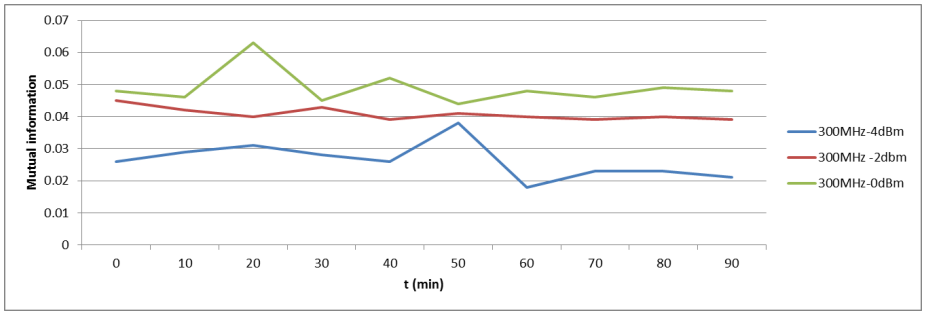
4.2.3 Mutual information

The design has 2 ROs implemented which can be monitored at the same time. Since this FPGA produces random numbers the mutual information will be close to zero without injection. When not injecting any signal, the mutual information is in the range $[0.001 - 0.09]$. Figure 4.4 shows the mutual information between the 2 implemented ROs when injecting a signal. When injecting on 220 MHz, the best mutual information is achieved when injecting with an input power of -2 dBm, varying between $[0.14 - 0.18]$. This means that an injection of 220 MHz on -2 dBm causes a common effect between the ROs, but statistical tests show it is not enough to bias the output of the TRNG. When injecting -4 or 0 dBm the mutual information is around $[0.04 - 0.08]$. This shows that more injection power does not necessarily yield better mutual information.

When injecting on 300 MHz, more input power yields a higher mutual information, although the mutual information is not a high value. When not injecting any signal, the mutual information is in the range $[0.001 - 0.09]$. Although 300 MHz was not chosen as a good frequency, it also proofs itself to be a bad injection frequency with almost no result for any of the input powers selected.



(a) Injection on 220 MHz



(b) Injection on 300 MHz

Figure 4.4: Mutual information for different input powers (-4, -2 and 0 dBm)

4.3 Conclusion

Injecting a signal causes a change of the operating frequency of the ROs in the FPGA. A higher injection power results in a lower operating frequency due to more rising of heat. A higher injection frequency can cause a higher operating frequency compared to injection of a lower frequency. Although the chosen frequencies were not chosen because they are optimal injection frequencies, it can be seen that injecting on 220 MHz does change the behaviour of the TRNG, with a maximum mutual information between the 2 ROs of 0.18. It is also shown that injecting more power does not mean a higher MI between the two ROs.

Chapter 5

TRNG implemented with 5 ROs

The amplifier used for these experiments was new. The effects induced by this amplifier was not yet investigated. Chapter 4 shows that the amplifier injects a signal and changes the behavior of the TRNG. This Chapter will describe a replication of the research performed by Bayon et al. in order to verify that the amplifier can bias a RO-based TRNG. We will first discuss the slightly different RO-based TRNG design compared to the research by Bayon et al., followed by a frequency sweep. Afterwards the mutual information will be shown, followed by some power sweeps on some of the best frequencies based on the mutual information. Then the test results of the NIST test suite will be shown, together with visual representations of the random number. The last section will be a conclusion based on these results. Appendix D describes some additional experiments performed with a smaller and isolated probe.

5.1 Design

Figure 5.1 shows the layout of the ROs of the TRNG for this experiment. There are 5 ROs implemented in this design, all placed horizontal with some space between the ROs. They are placed towards the left part of the chip (within the red circle) to prevent influencing other parts of the design (like the XOR-tree and the FIFO queue and the registers) during the injection. The top horizontal row is the first RO, the second row the second RO etc. Although there is some space in-between the ROs, it might still be possible that cross-talk happens between the antennas of the ROs. To be able to influence all the ROs it was chosen to keep the ROs close together, but far enough to prevent interlocking of the ROs without any injection. The implemented design passes all tests from the NIST SP-800 test-suite with a file of 1 GB.

The operating frequencies of the ROs of this design are shown in Table 5.1. As shown before, this operating frequency changes due to temperature and the

RO nr.	Frequency (MHz)
1	242
2	248
3	255
4	241
5	241

Table 5.1: Rough estimation of the operating frequencies of the ROs.

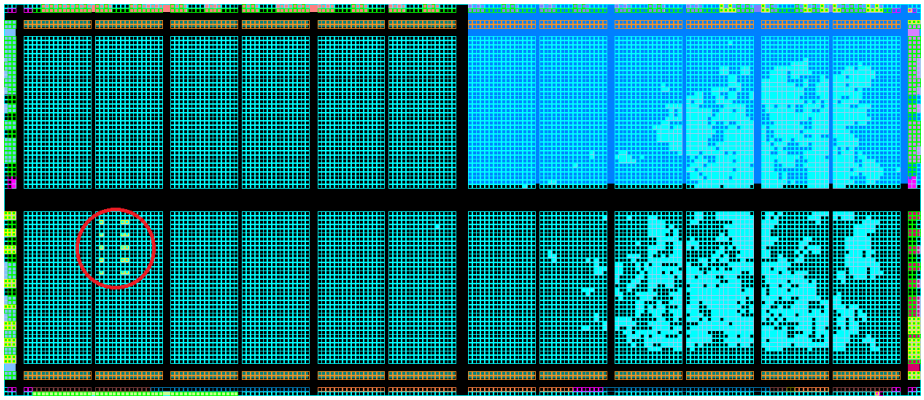


Figure 5.1: Design consisting of 5 ROs of length 3, all with an antenna

introduced fault injection.

5.2 Frequency sweep

To find the optimal injection frequency a frequency sweep was done. Section 3.4.1 describes 2 methods to finds the optimal injection frequency. The first method divides the intensity of the injection peak by the intensity of the peak of the operating frequency of the RO. The second method substracts the intensity of the peak of the operating frequency during injection by the intensity of the peak of the operating frequency without injection. Both methods will be discussed below.

Figure 5.2 shows the results for dividing the peak of the operating frequency of the RO by the injection peak. Figure 5.3 shows the sum of all the results of the ROs. The left side of the frequency spectrum has no to little effect, but there is an optimum towards the right side of the spectrum. After 278 MHz the optimum seems to decrease again. Table 5.2 shows the top 10 optimum injection frequencies according to this method.

Table 5.3 shows the top 10 best results for the second method. There seems to be some optimum around 257 MHz to 259 MHz. The best injection frequency

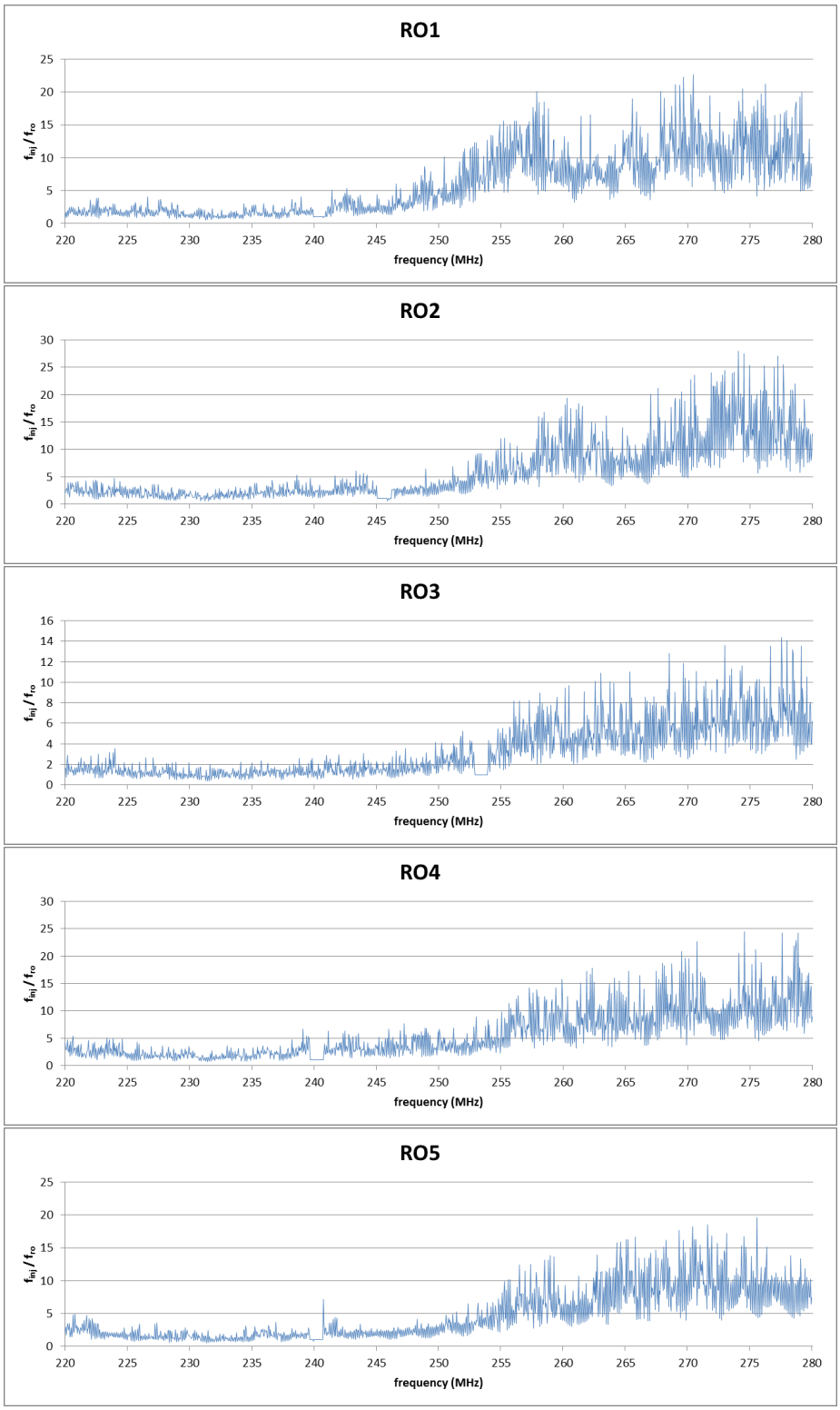


Figure 5.2: $\frac{dB(f_{inj})}{dB(f_{RO})}$ for every implemented RO of the TRNG

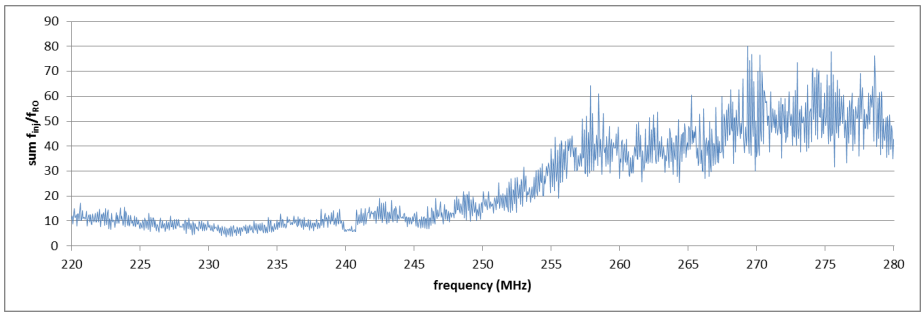


Figure 5.3: Sum of the $\frac{dB(f_{inj})}{dB(f_{RO})}$ values

Nr.	f_{inj}	$\frac{dB(f_{inj})}{dB(f_{RO})}$
1	269.3	80.1646
2	275.4	78.02387
3	269.6	76.79103
4	270.2	76.65576
5	278.6	76.32558
6	269.45	74.50722
7	272.95	73.65841
8	274.05	71.45237
9	274.35	70.80369
10	274.5	70.36323

Table 5.2: Top 10 best results for dividing the injection peak by the peak of the operating frequency of the RO.

is 248.6 MHz, which is the exact operating frequency of RO2 at that time. The 9th best injection frequency is 255.45 MHz, which was the operating frequency of RO3 at that point in time. The 3rd best injection frequency is 269.6 MHz, which was also the third best injection frequency in the previous methods. When going through a larger set of the results, there seems to be more overlap between the different verification methods (e.g. an injection of 257.85 MHz is the 25th best injection frequency for the first method and the second best injection frequency for the second method).

Several candidates for an optimal injection frequency have been chosen and are listed below:

- 228.5 MHz
- 248.6 MHz
- 257.85 MHz
- 269.3 MHz
- 269.45 MHz
- 269.6 MHz
- 275.4 MHz

Nr.	f_{inj}	$dB(f_{ROinj}) - dB(f_{RO})$
1	248.6	-25.6037
2	257.85	-24.628
3	269.6	-22.2255
4	235.25	-21.3734
5	228.5	-18.2788
6	258.5	-17.7511
7	258.85	-16.385
8	228.05	-16.3544
9	255.45	-15.6487
10	258.8	-14.8368

Table 5.3: Top 10 best results for substracting the RO peak during injection by the RO peak without injection

The best three injection frequencies for both methods were chosen. In addition to these, the injection frequency 269.45 MHz was chosen because it appears in the top 10 and is exactly in between 269.3 MHz and 269.6 MHz, which are both in the top 3 in the first method. Furthermore, injection of 228.5 MHz produces 5 negative values for the second method, implying that all the ROs have locked to a different frequency and might have interlocked. The next Section will continue with a small power sweep for the different candidates. During these power sweeps, the MI between two ROs was measured.

5.3 Mutual information

A small power sweep was performed on the previously described candidates as optimal injection frequencies. During the power sweep the mutual information between 2 ROs was calculated. The power sweep ranged from -2 dBm to 0 dBm, with a step size of 1 dBm. This was done for every pair of ROs (thus 10 measurements for every different input power injection). Figure 5.4 shows the average result of all the power sweeps on the different input power injections. Figure 5.4 also shows some initial mutual information of 0.1 to 0.2. This is because RO1, RO4 and RO5 seem to be interlocking without injection taking place. This could be caused by the fact that their operating frequencies are close to each other and some cross-talk between the introduced antennas in the ROs. Although the ROs are not completely independent, the output of the TRNG was still statistically random.

Figure 5.4 clearly shows that injection of a frequency at -2 dBm increases the mutual information between the ROs with respect to no injection. For all the chosen injection frequencies the MI goes up during injection compared to no injection. The best injection frequency (of the chosen injection frequencies) seems to be 257.85 MHz, reaching a maximum MI of 0.46 bits. From Figure 5.4 it also seems like more power does yield a higher MI, although this shouldn't

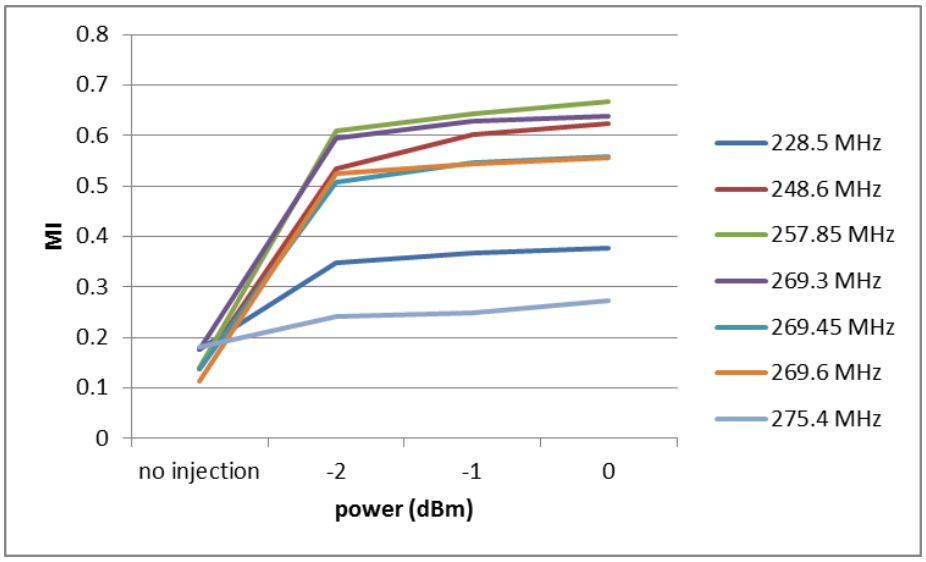


Figure 5.4: Averages of every MI between 2 ROs for all the chosen frequencies

necessarily be the case (see Figure 4.4).

5.4 Power sweep

A larger power sweep was performed on an injection of 257.85 MHz. The power sweep ranged from -8 dBm to 0 dBm with a step-size of 1 dBm. Figure 5.5 shows the MI for every couple of ROs. As can be seen, RO1 and RO5 are already interlocked with an MI of 0.73 without injection. Nevertheless, injection does increase the MI between the two ROs even more. Getting a high mutual information between RO2 and RO3 appears to be the most difficult objective, although a high MI between RO2 and RO3 also seems hard. The chosen injection frequency seems to lock most of the ROs, but seems to be less effective for RO2 in combination with RO3 and RO4.

Injection with an input power of -4 dBm seems to enhance the MI between RO1 and RO5, but decreases the MI between RO1 and RO4. It therefore seems that different powers might work better for 2 ROs, while performing worse for others. In general, more power does seem to increase the MI between 2 ROs. Since it was shown in Figure 4.4 that this is not necessarily true, the measured MI started to become doubtful. It could be possible that the LeCroy probes measured the signal of the injected signal over the air in conjunction with the signal of the ROs. However, this noise coming over the air should not have an significant effect on the measured signal of the ROs. Nevertheless, we decided to shield the LeCroy probes with some aluminum foil for the next experiments to prevent measuring this noise as much as possible.

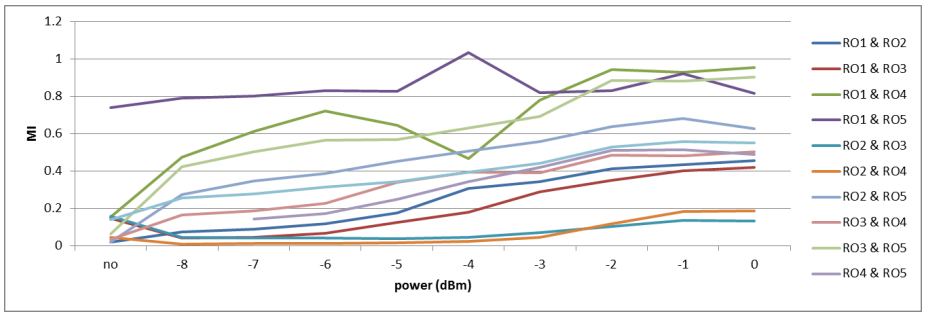


Figure 5.5: The MI during a power sweep on 257.85 MHz

5.5 RNG test suite result

A file of 1 GB of the output of the TRNG was gathered during injection of 257.85 MHz with an input power of -2 dBm. Note that it passed all the NIST-tests without any fault injection with the same size. The TRNG failed all the NIST-tests (including the monobit test) with the file that was taken during the injection. However, although 1.4% of the random number was biased in the monobit test, a visual inspection lacks a result comparable with those shown by Bayon et al.. Figure 5.6 shows the maximum bias as the visual result, with the bit zero drawn as a white square and the bit one drawn as a black square. The number drawn consists of 3840 (60 x 64) bits. The maximum bias achieved towards ones is 7.3% (2062 of the total amount of 3840 bits are 1's) and the maximum bias achieved towards zeroes is 8.0% (2074 bits of the total amount of 3840 bits are zeroes). For comparison, Figure 5.7 shows the results achieved by Bayon et al. at different output powers (PForward). This PForward is the power emitted by the probe and is a different power than the input power used in this thesis, as explained in Chapter 3. The research done by Bayon et al. achieved a bias of 55% towards zeroes. It might be possible that the injected signal is directly picked up by the LeCroy probes, instead of propagated over the signal of the ROs.

5.6 Conclusion

This experiment was a full replica of the research performed by Bayon et al. Although a high mutual information was achieved and the TRNG becomes biased (failing the NIST monobit test), a visual results like those presented by Bayon et al. were not achieved. Although the visual results are not comparable with those achieved by Bayon et al., good confidence was found in the MI which shows that the ROs were mutually interlocked. The following experiments will have an aluminum foil wrapped around the LeCroy probes, to prevent them from measuring the injection signal as noise over the air. This should make it pick up none to low noise of the injection directly onto the LeCroy probe. The

(a) (b)

Figure 5.6: Visual representation (0's in white, 1's in black) of the maximum bias towards ones (Figura (a): 7.3%) and towards zeroes (Figure (b): 8.0%)

Figure 5.7: Results achieved by Bayon et al. [6]

rest of the research will continue to base its results on MI.

Chapter 6

Injection on different implementation designs

Although the experiment described in Chapter 5 showed promising results, reproduction of the experiment seemed to be hard and sometimes impossible. Therefore, to be able to study the exact effect of the FI on the ROs, we decided to implement a TRNG using only 2 ROs. This does not only make it possible to monitor all the entropy sources of the TRNG, but also makes the experiments more time-efficient. Calculating the MI of a TRNG using only 2 ROs requires only 1 measurement, while a TRNG of 5 ROs requires 10 measurements (one for each pair of ROs). The first section will describe the initial experiment performed, building up to the main research described in Section 6.2. Section 6.2 will describe possible designs in detail with the experiments performed on them. The sections afterwards will go into more detail for the designs and discuss the results.

6.1 Initial experiment

Having a design of 2 ROs makes it possible to calculate the MI during the frequency sweep, thus skipping the step to find the optimum injection frequency. The next experiment did a frequency sweep and analyzed the MI to find a good injection frequency. This experiment had an initial injection position slightly away from the location of the ROs (unintentionally) and an unexpected result with a maximum MI of 0.1. A second sweep was done, positioned right on top of the ROs, yielding better results. It seemed that the location was influencing the result. Therefore, 4 additional frequency sweeps were done. The first frequency sweep was performed on top of the ROs. The other four frequency sweeps were located in the corners of the chip.

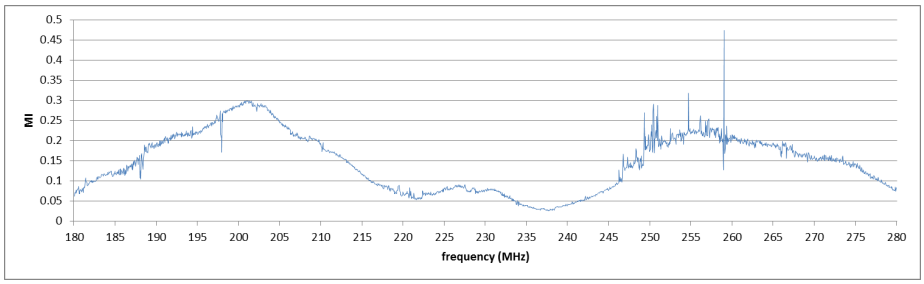


Figure 6.1: Frequency sweep located on top of the ROs

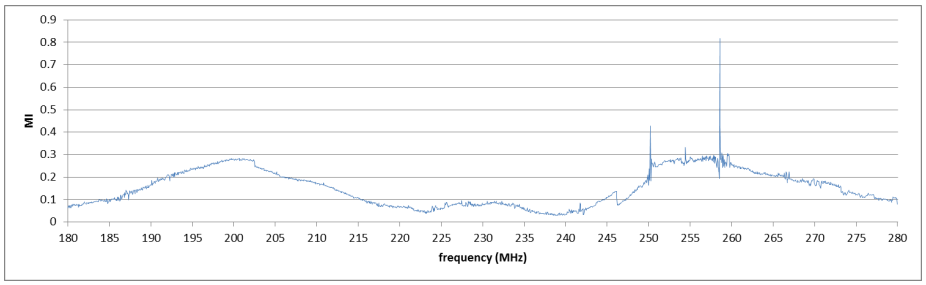


Figure 6.2: Frequency sweep located in the top left corner of the chip

6.1.1 Frequency sweep & mutual information

The frequency sweep ranged from 180 MHz to 280 MHz with a step-size of 50 KHz and a injection input power of -2 dBm. Figure 6.1 shows the frequency sweep with the probe located on top of the ROs. Figure 6.2 shows the frequency sweep with the probe located in the top left corner of the chip. The other frequency sweeps can be found in Appendix C. From Figure 6.1 it can be seen that there is some optimum around 202 MHz after which the MI starts dropping. After 237 MHz the MI starts to increase again, with 5 peaks. The peaks are at 249.3 MHz, 250.45 MHz, 250.95 MHz, 254.7 MHz and 259 MHz. From these 5 peaks, at the time the measurements were taken the second peak corresponds to the operating frequency of the first RO, the fourth peak corresponds to the mean of the operating frequencies of both ROs, and the fifth peak corresponds to the operating frequency of the second RO. Figure 6.2 shows similar behavior as Figure 6.1. Some optimum at 202 MHz and several peaks towards the right. At the time those measurements were taken, the first RO had an operating frequency of 250.2 MHz and the second RO had an operating frequency of 258.55 MHz. The first peak is indeed on the operating frequency of the first RO, the smaller peak in between the larger peaks is the mean of the frequencies, and the largest peak is the operating frequency of the second RO. All measurements (see Appendix C) show this behavior, with a peak at the operating frequency of the RO and the mean of the operating frequencies.

It seems that injecting on the operating frequency of the RO seems to be a good way to achieve a high mutual information. Also, the value of the MI differs between the different injection locations. Injecting on top of the ROs achieved a maximum MI of 0.47, while injecting in a corner (outside of the programmable die of the chip, see Figure 6.3) achieves a maximum MI of 0.82. This shows that injecting with the operating frequency of the implemented ROs seems to be a very efficient way to interlock the ROs, but the maximum value of MI also seems to depend on the location of the injection. Note that the MI is purely based on the signal of the ROs, since the probes are shielded and should not measure any noise (or at the least should have a very low noise-level).

6.2 Designs

To make it easier to determine the effect on the ROs, several designs were chosen of a TRNG with only 2 ROs. Although the introduced antenna in the ROs is assumed to enhance the effect, it is not yet investigated to be true. This section will thus also aim to elaborate on the effect of the antenna. Since the effectiveness of the injection also seems to be dependent on the location of the injection, an area to do an XY-scan is defined first.

6.2.1 Scanning area

In order to see the effect of different locations of injection, an area was defined to perform an XY-scan on. To know what area to scan, the actual size of the die of the chip needs to be known. Since the FPGA had to be returned to the previous owner, decapitation of the chip was not an option. Therefore a design was created with 2 different ROs with 2 very different operating frequencies. One RO was put in the bottom left corner and one RO was put in the upper right corner. Hovering over the chip with an EM-probe can pick up the signal emanating from the chip. The LeCroy can calculate the FFT spectrum from that signal on the spot and by looking at the spectrum it is known where the RO is located. Although this method is not very precise, it does give an approximation of the size of the programmable die. Although there is more in the chip than just the programmable die, it does give an approximation on the point of injection and might give some insight in the areas the FI is affecting.

The approximate size of the programmable die is shown in Figure 6.3(blue rectangle). The scan was performed on the chip which was rotated 90 degrees counter-clockwise. The chip was also mirrored (or up-side-down). The ROs were placed in the bottom left corner (blue dot in Figure 6.3), and the scan area was a 15x15 grid around the placement of the implemented ROs (red rectangle in Figure 6.3). Note that these areas are not precise (both the programmable die area as the scan area), but is merely an indication of the areas that will be talked about in the next experiments. The XY-scan starts from the top left corner towards the bottom right corner. The ROs will be located near the 8th row and the 5th column in the scanned grid-area.

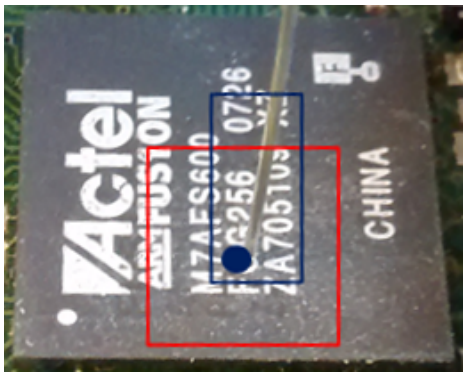


Figure 6.3: The chip and its programmable die (blue rectangle), location of the ROs (blue dot) and the scan area (red rectangle)

6.2.2 The chosen designs

There are several ways to implement a TRNG based on ROs. A design that implements the ROs in a horizontal orientation results in different frequencies than a design that implemented the ROs in a vertical orientation. This difference in operating frequency can be as large as 20 MHz. An artificial antenna was also introduced between the first and second element of the RO. The effect of this antenna can also be tested in comparison with a design that does not have this antenna. The way the ROs are organized can also influence the output of the TRNG. If ROs are placed parallel to each other, crosstalk might occur between the ROs. A design that has the ROs in-line might suffer less from the crosstalk. Another design choice is to change the length of the ROs. For the next experiments, the designs that have an unequal ring length will consist of 3 inverters for the first RO and 5 inverters for the second RO. For every design that has an unequal length in combination with an antenna results in an antenna for the first (3 inverter) RO only. The main reason for this design choice is that the antenna results in operating frequencies that are closer together. The surface of a RO of length 5 is also bigger and almost equal to that of a RO of length 3 with an antenna. The list below summarizes the different designs that will be tested and will be elaborated in the next sections. All these different designs are done in combination with each other, resulting in (2x2x2x2=) 16 different designs.

- The different between a horizontally placed ROs and vertically placed ROs.
- With and without an antenna between the first and second element of the RO.
- Placing the ROs parallel to each other or in sequence with each other.

- ROs of different ring length, where one RO has a length of 3 elements in and the second RO has a length of 5 elements.

6.2.3 Flow of an experiment

For each design mentioned, an XY-scan over an area of the die of the chip as displayed in Figure 6.3 (red rectangle) is performed. The area is divided into parts of 15 by 15 and thus gives 225 measurements per experiment. The area is scanned three times for one injection frequency and 3 injection frequencies were chosen. The injection frequency is dynamically calculated and corresponds to the frequency of the first RO, the frequency of the second RO and the mean frequency of the ROs.

Every experiment followed the following procedure:

1. Acquire two traces from the LeCroy, corresponding to a trace for the first RO and the second trace for the second RO. These traces are taken without any injection performed.

2. From the 2 gathered traces from step 1, calculate the FFT for each. The operating frequency of the first RO can be calculated from this first FFT trace. The operating frequency of the second RO can be calculated from the second FFT trace. The operating frequency of the RO is the x-value where the highest peak of the trace is. Based on the chosen mode, the injection frequency is chosen. This is either the operating frequency of the first RO, the operating frequency of the second RO or the mean of both frequencies. The injection frequency is communicated to the signal generator and the amplifier is turned on.

3. Acquire two traces from the LeCroy, where the first trace corresponds to the first RO and the second trace to the second RO. These traces will be taken while there is an injection taking place.

4. Turn the amplifier off and go to the next position of the XY-scan. Repeat the process from step 1.

Due to temperature changes and the injection done for the previous location, the frequency of the ROs changes. This is the reason why the frequency of the RO needs to be calculated again for every location. The temperature changes might be environmental changes (e.g., heat emitted by the amplifier), but also temperature changes induced by our injection which causes a higher temperature in the chip itself because of electric coupling. This workflow makes it possible to inject a frequency that is close to equal to the operating frequency of either RO.

For every experiment, the mutual information is calculated and put into an XY-plot. The different coloring of the squares represents the amount of mutual information. The scaling of these colors will be elaborated on in each experiment. A dark red square represents the highest MI achieved during the experiment. A dark blue square represents the lowest measured MI.

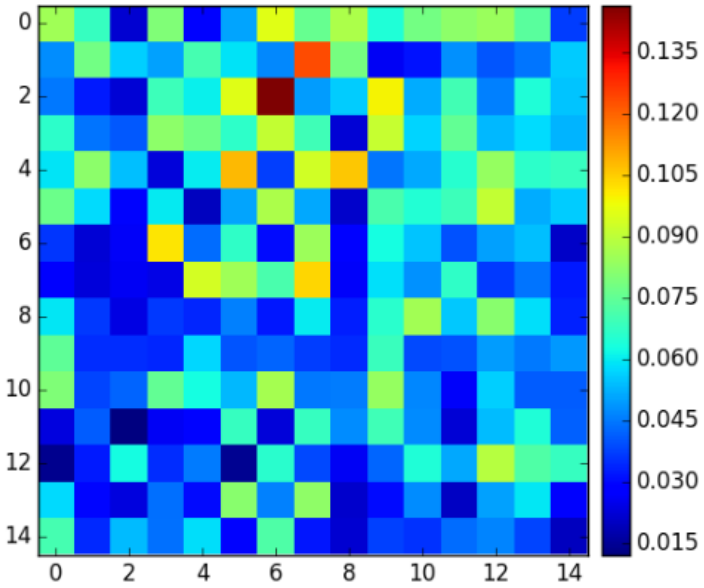


Figure 6.4: Horizontal, without antenna, in-line, equal length. Average of 3 XY-scans with an injection of the frequency of the second RO

6.3 General remarks

The average of the 3 XY-scans for an injection frequency for every design can be found in Appendix E. Appendix E will discuss every design separately. This section will continue with some general remarks and give some conclusion on the effects of using the different designs.

Figure 6.4 shows the average result of three XY scans with an injection of the operating frequency of the second RO. The maximum achieved MI is 0.138 at spot (2,6). The ROs are located at spot (8,5) and the maximum is thus not reached on top of the ROs. However, spots with some of the highest MI during this experiment, namely (2,6), (4,5), (1,7) and (4,8), are all located near the wire that goes from the third inverter element to the XOR-gate. Influencing the wire going to the harvesting mechanism can thus also affect the ROs and propagates the injected signal backwards into the ROs.

Another result of the average of a frequency injection equal to the operating frequency of the second RO is shown in Figure 6.5. The ROs in this design were placed horizontal, without an antenna, inline of each other and of unequal length. This result also has some high MI values near the wire going towards the harvesting mechanism (from (8,5) to (0,9)). However, there are also high MI values in the 10th and 11th row. These high MI values might give a wrong idea, since this could be an influence on the bonding wire going from the selected output pin in the chip towards the output pin to which the LeCroy probes are

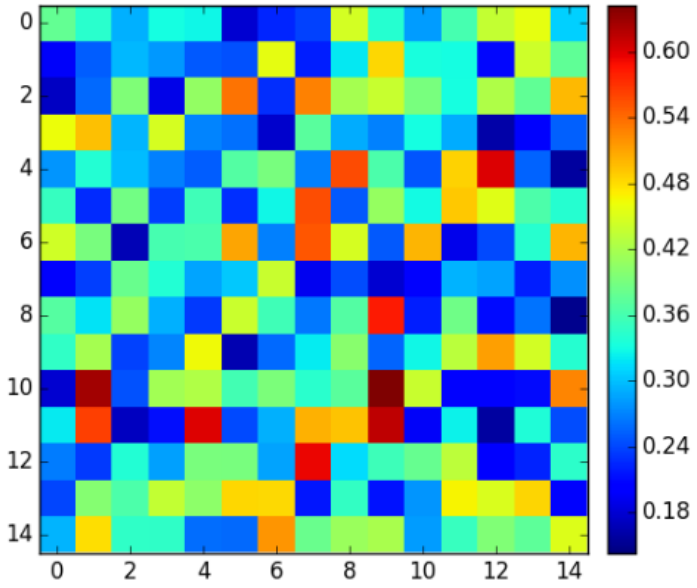


Figure 6.5: Horizontal, without antenna, in-line, unequal length. Average of 3 XY-scans with an injection of the frequency of the second RO

attached. It can also be possible that these bonding wires were influenced and the signal propagated backwards into the ROs (as is the case for the wire going to the harvesting mechanism) and did influence the ROs. A definitive conclusion on this cannot be given, but could be tested by monitoring and testing the output of the TRNG.

Another conclusion drawn from Figure 6.5 is that the location of the injection matters a lot. An injection with the operating frequency of RO2 at spot (4,12) results in an average MI of 0.6, while a location slightly above it ((3,12)) yields a MI of 0.1 and is thus ineffective. The importance of this location dependent success is one of the reasons for not being able to replicate results from previous experiments. Indeed, before doing the experiment of Section 6.1 it was believed that the effect of an injection was a global effect. This research shows that this is not the case and that it is a local effect when injecting a frequency equal to the operating frequency of the ROs. When injecting the mean of the operating frequencies, it does seem to be a more or less global effect.

The ROs of unequal length are believed to produce better results since the frequency difference is larger and mutual interlocking is harder to achieve. Although this design does not produce a statistical random number, it does pass the monobit-test from the NIST test-suite. Also, the mutual information between the 2 ROs is between 0.01 and 0.18 without injection, while the mutual information for equal length ROs is between 0.01 and 0.09.

Using a design that has vertically parallel placed ROs of equal length without

an antenna seems to be the most insusceptible to this attack, with a maximum MI of 0.08, which is equal to some measurements without injection. This design does show that the top right corner of the scanned area is influenced more than the rest of the chip. Although no definitive reason for this effect can be given, there might be something that is connected to the ROs. It could be possible that there might be a power supplier for the FPGA near that spot, which might propagate the signal over the power net into the ROs. Unfortunately this cannot be proven and is merely one guess among many possible options.

Injecting on the operating frequency of RO2 seems to be more effective than injecting the operating frequency of RO1. Exception to this rule are the designs placed horizontal, with antenna in parallel of unequal length and vertical with antenna, inline and unequal. Common factor between these designs is the unequal ring length and both have an antenna. For the second design however, the MI went up to 0.42 which is also a good result. The first design had a MI of 0.09 when injecting the operating frequency of RO2, which can be seen as ineffective.

6.4 Horizontal vs vertical placement

The assumption is that the placement of the RO in a horizontal or vertical orientation will not differ in the results. This seems to be correct, although there might to a relation with the implemented antenna. The next section will discuss this relation.

6.5 With antenna vs without antenna

We assume that the antenna introduced between the first and second element of the RO will yield better results. As pointed out in Section 6.3, the wire going to the harvesting mechanism can also act as an antenna. The wire going to the harvesting mechanism is longer than the wire introduced inside the RO(s). Although electric coupling might be less on the lower frequency resulting from the antenna between the first and second element of the RO, it is still believed that this antenna should yield better results. The experiments show that this is not always correct. For the design that had the ROs placed vertical with an antenna generally had a higher result than the designs without an antenna. The opposite is true for horizontal placed ROs, where horizontal designs without an antenna generally achieved higher MI than those with an antenna. There is an exception to this, namely the design with ROs placed vertical, parallel and an unequal length. In that case the design without the antenna performs better when injecting the mean operating frequency of the ROs. Table 6.1 shows the maximum achieved average for each design.

Table 6.1 also shows that the use of an antenna in a RO of unequal length does not make it more influencable. Although RO1 (the RO of length 3) was the only RO with the antenna, these cases do not always show a higher MI than

the same design without an antenna (and vice versa).

	with antenna				without antenna			
	f_{RO1}	f_{RO2}	f_{mean}	average	f_{RO1}	f_{RO2}	f_{mean}	average
horizontal, inline, equal	0.08	0.11	**0.12**	0.10	0.09	**0.15**	0.06	0.10
horizontal, inline, unequal	0.34	**0.36**	0.07	0.26	0.31	**0.64**	0.37	0.44
horizontal, parallel, equal	0.14	**0.21**	0.09	0.15	0.19	**0.23**	0.07	0.16
horizontal, parallel, unequal	**0.52**	0.09	0.07	0.23	0.26	**0.49**	0.33	0.36
vertical, inline, equal	0.18	**0.37**	0.20	0.25	0.05	**0.18**	0.03	0.09
vertical, inline, unequal	**0.72**	0.42	0.06	0.40	0.08	**0.57**	0.07	0.24
vertical, parallel, equal	0.23	**0.46**	0.21	0.30	0.01	**0.09**	0.01	0.04
vertical, parallel, unequal	0.36	**0.55**	0.07	0.33	0.19	0.44	**0.65**	0.43

Table 6.1: Design comparisons between ROs with antenna and without antenna based on MI

6.6 Inline vs parallel placement

Parellel implemented ROs can have some crosstalk between the ROs, while ROs that are placed in-line suffer less from crosstalk. We therefore assume that parallel placed ROs have a higher MI than ROs placed inline. It turns out that the amount of crosstalk between the 2 ROs in these experiments do not have a huge impact. Results are somewhat comparable between the ROs that are placed parallel and the ROs that are placed in-line. Sometimes parallel ROs have a higher MI, sometimes in-line placed ROs have a higher MI. Injection of the mean frequency of the operating frequencies seems to have a constant result between parallel and in-line placed ROs. There is one design that is an exception to this, which is the vertically placed, without antennas and unequal length ROs. Injection of the mean frequency has a higher impact on parallel placed ROs than in-line placed ROs for that design.

6.7 ROs of different length

As stated in Section 6.3, the designs with ROs of different length do not produce statistical random numbers and already have a slightly higher MI without injection compared to the equal length ROs without injection. Table 6.3 lists the

	inline				parallel			
	f_{RO1}	f_{RO2}	f_{mean}	average	f_{RO1}	f_{RO2}	f_{mean}	average
horizontal, with antenna, equal length	0.08	0.11	**0.12**	0.10	0.14	**0.21**	0.09	0.15
horizontal, with antenna, unequal length	0.34	**0.36**	0.07	0.26	**0.52**	0.09	0.07	0.23
horizontal, without antenna, equal length	0.09	**0.15**	0.06	0.10	0.19	**0.23**	0.07	0.16
horizontal, without antenna, unequal length	0.31	**0.64**	0.37	0.44	0.26	**0.49**	0.33	0.36
vertical, with antenna, equal length	0.18	**0.37**	0.20	0.25	0.23	**0.46**	0.21	0.30
vertical, with antenna, unequal length	**0.72**	0.42	0.06	0.40	0.36	**0.55**	0.07	0.33
vertical, without antenna, equal length	0.05	**0.18**	0.03	0.09	0.01	**0.09**	0.01	0.04
vertical, without antenna, unequal length	0.08	**0.57**	0.07	0.24	0.19	0.44	**0.65**	0.43

Table 6.2: Design comparisons between ROs placed inline and ROs placed in parallel based on MI

highest MI of the average of the 3 XY-scans for every design. We assume that a high MI between ROs of unequal length was harder to achieve compared to ROs of equal length. However, results show that injection on designs of unequal length seem more susceptible to injection. For every design the design with ROs of unequal length design achieved a higher MI than the same design with ROs of equal length.

6.8 Conclusion

The experiments on different designs show that injection of the operating frequency of an RO is location dependent, although injection of the mean of the operating frequencies of the ROs seems to be global. The designs with ROs of

	equal length				unequal length			
	f_{RO1}	f_{RO2}	f_{mean}	average	f_{RO1}	f_{RO2}	f_{mean}	average
horizontal, with antenna, inline	0.08	0.11	**0.12**	0.10	0.34	**0.36**	0.07	0.26
horizontal, with antenna, parallel	0.14	**0.21**	0.09	0.15	**0.52**	0.09	0.07	0.23
horizontal, without antenna, inline	0.09	**0.15**	0.06	0.10	0.31	**0.64**	0.37	0.44
horizontal, without antenna, parallel	0.19	**0.23**	0.07	0.16	0.26	**0.49**	0.33	0.36
vertical, with antenna, inline	0.18	**0.37**	0.20	0.25	**0.72**	0.42	0.06	0.40
vertical, with antenna, parallel	0.23	**0.46**	0.21	0.30	0.36	**0.55**	0.07	0.33
vertical, without antenna, inline	0.05	**0.18**	0.03	0.09	0.08	**0.57**	0.07	0.24
vertical, without antenna, parallel	0.01	**0.09**	0.01	0.04	0.19	0.44	**0.65**	0.43

Table 6.3: Designs comparison between ROs of equal length and unequal length based on MI

unequal length are more susceptible to an injection of the operating frequency of a RO or the mean of the operating frequencies of the ROs. Furthermore, designs with vertically placed ROs with an antenna generally had a higher result than the designs without an antenna. Designs without an antenna placed horizontally have a higher MI than the horizontal designs with an antenna. The designs containing the ROs in-line seems to have similar effects compared to the designs where ROs are implemented in parallel.

Chapter 7

Conclusion

The research question that this research seeks to answer is:

- Is an EM-FI using harmonic emission attack on a different length RO-based design feasible?

In order to answer the research question, additional experiments were performed. A replication of the research performed by Bayon et al. was done. The only part that this research was not able to replicate was the visual representation of a biased random number. This research did replicate a successful injection that resulted in the output of the TRNG fail the NIST-monobit test. Achieving the biased result reported by Bayon et al. is not straightforward in our case.

This research shows that an injection frequency equal to the operating frequency of one of the ROs or the mean of the operating frequencies of the ROs is a good injection frequency that results in high mutual information. However, when injecting with a frequency equal to the operating frequency of one of the ROs, the location seems to be important. The best location to inject is either slightly next to the ROs, or (if applicable) a long wire connected to the ROs. In this research the wire going into the harvesting mechanism seemed to be a good injection location. The bonding wire that connected the signal of the RO to an output pin also propagated the fault injection, although it can not be proven that this signal is propagated backwards into the ROs.

Sixteen designs were tested, varying in orientation of the ROs, with or without a long wire between the first and second element (acting as an antenna), parallel or inline of each other, and equal or unequal ring length. Vertically placed ROs with an antenna seem to be more susceptible to this kind of attack compared to vertically placed ROs without an antenna. On the other hand, horizontally placed ROs without an antenna seem to more susceptible to an EM-FI-HE attack than horizontally placed ROs with an antenna. Whether ROs were placed in parallel or in-line of each other does not give a distinctive result.

Although the output of the TRNG using ROs of unequal length was not random without injection, this research shows that ROs of unequal length seem to be more susceptible to this attack compared to TRNGs using ROs of equal length for the created designs. The MI during injection on ROs of unequal length went up to 0.92, although the point of injection is important. A less effective spot could yield a MI less than 0.1.

This research bases its conclusion merely on the achieved MI and not on the output of the TRNG. It could be possible that the injection affected bonding wires connecting the RO to an output pin. A high MI could thus be caused by influencing this wire and does not necessarily mean a successful attack on the TRNG, although the FI could propagate into the ROs.

The measured MI is an indication of the quality of the output of a TRNG based on ROs. Although the random number tests show that the output is biased, a visual representation of the number did not show it. This research can therefore not indicate if this effect is sufficient enough for a realistic attack on a security device.

Chapter 8

Future work

Chapter 7 states that the results reported by Bayon et al. can only partly be replicated. We show that the TRNG fails the monobit test, but a visual representation as shown by Bayon it al. cannot be achieved. One reason might be the difference in power injected by the probe. To be able to compare this research with the research performed by Bayon et al., the power emitted by the probe should be measured. In the current setup the power put into the amplifier is known, but we cannot quantify the power that is emitted by the probe. Although the MI is comparable to those presented by Bayon et al., the power used to get a 50% biased output from the TRNG is a lot less than those used to calculate the MI. The authors of the paper by Bayon et al. stated in a discussion that more emitted power might actually counter the effects. As a reason they give the influence on other parts of the system. Using more power makes it more likely to influence different parts of the FPGA, like the sampling mechanism (D-flipflop). A D-flipflop gives the input as output every time the clock-signal transitions from low to high. An EM-FI-HE attack might influence this clock signal, making the D-flipflop give output more often than expected.

Despite a lack of being able to measure the power emitted by the probe, this research defines good injection frequencies and elaborates on good injection spots. However, the effect of different input powers is rather unexplored. If the power emitted by the probe cannot be measured, a power sweep can also be done. Although this research did some power sweeps, a more detailed power sweep should be done. Bayon et al. show that the result can differ in a range of 50 μW (the power emitted by the probe). The range of the power sweep should be larger than presented in this research. If it is true that more power counters the effect, a power sweep should investigate the effect of lower input powers, preferably with smaller steps than presented in this thesis.

In addition, this research showed that high MI is achieved, but that visual results lack. The lacking of this visual results is partly because there was no system to have a visual representation of the number during injection. Visual results could only be viewed after the experiment was done. A 'live' update of the output of the TRNG might be useful during a previously mentioned more

detailed power sweep to make it more easy to find a good injection power.

This research had trouble with reproducing some of the results. As explained, this research measured the output of ROs using a LeCroy probe. These probes should perform good on high frequencies like those in this research. However, it was also found that a small alteration in the positioning of the cable (such that it is measuring more or less power) can increase or decrease the MI by 0.05. Therefore it would be preferable to have these cable fixed in some way (e.g. sticked onto the table).

Another reason for reproducibility issues might be the change in temperature. The amplifier becomes heated after several minutes of injecting. This heat is dealt with by fans, blowing out the hot air. This hot air is blown towards the FPGA, which heats the chip and thus changes the operating frequencies of the ROs. Some regulations in the temperature might be nice to have, although a stable temperature inside the FPGA could be hard to maintain. Temperature regulation of the outside of the FPGA could give less changes in frequencies of the ROs. It is not known however if the operating frequency of the ROs mainly changes because of a hotter temperature outside of the FPGA or because the internal heat of the FPGA went up due to electric coupling induced by the injected signal.

Repeating this experiment with the smaller probe can give more insights in how the ROs are affected by the FI. This research only did one frequency sweep with a short probe, but did not follow up on the results. These results can be found in Appendix D. Because the short probe was isolated, it was believed that the injection signal was more 'aimed' towards the FPGA. Although this research did not follow up on this theory, it seems like an efficient way to influence a small part (e.g. ROs) of the architecture on the FPGA.

Even if the described attacks yield successful results, the attacks shown in this thesis are all performed on a white-box target. However, a high-end security target is usually a black-box. A good point for injection could be hard to find in a black-box target. The same applies for finding the operating frequencies of the ROs. Even if these are successfully found, a post-processing algorithm like von Neumann counters this attack partly. The XOR-tree of the harvesting mechanism should produce a lot of zeroes if this attack is successful. Using von Neumann as a post-processor will not succeed in biasing the TRNG, but will result in a denial of service (since von Neumann discards two adjacent 0's). This can be detected in the output rate of the TRNG, making the target aware that it is being tampered with.

Appendix A

Additional TRNGs

To get some more insight in how TRNGs are constructed, some more TRNGs and the reason why they are truly random are described below. These TRNGs had no relation to the research performed.

A.1 Quantum optical

When creating a truly random number generator, it makes sense to use existing theories that are inherently random. Using the quantum theory therefore is an obvious choice, since the quantum theory predicts that each individual choice is truly random and independent of another choice. There are 2 TRNGs methods that employ the quantum theory in their design. One method is the quantum optical theory (discussed below), the other is the radioactive decay (Section A.2).

There are three ways of retrieving a random number out of quantum optics. First, it is possible to use photon detection times. Second, you can measure the polarization of photons. Third, you can combine both. Using optics is also a way to get a high output rate (some report an output rate of 140Gb/s).

A.1.1 Photon detection time

Stefanov et al. [33] describe a 'Optical Quantum Random Number Generator'. It consists of a pulser (830 nm LED) which is coupled into a monomode fiber (fiber which lets light go in only 1 direction). The photons at the end of the monomode fiber are indistinguishable at this point. There are two paths to get to the photon detector, one path labeled as '0' and the other path labeled as '1'. The path labeled as '1' has a 60 ns delay, which makes it possible to verify which path the photon took by the detector. The downside of this approach is that it cannot use a continuous laser, but needs a light pulse.

Dynes et al. [13] solve the disadvantage of Stefanov et al. Dynes et al. describe a system that can use a continuous laser and does not need any post-

processing to retrieve a random bit string. The system uses a weak photon induced avalanche, obscured by an avalanche photodiode. The signal of the avalanche photodiode is send over two wires, where one wire is one clock cycle longer than the other. The signals are then subtracted from each other, leaving the weak avalanche signal (also shown by Yuan et al. [43]). This remaining signal is amplified and send into a time tagging single photon counting electronic. If a detection took place in an even clock cycle the output would yield a '1'. If a detection took place in an uneven clock cycle the output would yield a '0'.

A.1.2 Polarization

Polarization based on two photons is described by Hai-Qiang et al. [15]. They were one of the first to describe a method with a continuous laser instead of a light pulser. A laser was aimed at a crystal and reached a polarized beam splitter. This polarized beam splitter reflected signal photons, but transmitted idler photons. These idler photons reach a detector, which signals the arrival of a signal photon at a 50:50 beam splitter. The signal photon follows a path, reaching either detector B or detector C. As output, there are four states. Either detector B detected something and detector C did not ($1_B\,0_C$), vice versa ($0_B\,1_C$), both detectors detected a photon ($1_B\,1_C$) or both detectors detected nothing ($0_B\,0_C$). The third state should not occur ideally, but could be possible due to stray photons or noise. The fourth state indicates a low photon flux and low detection efficiency. The other 2 states are being post-processed by means of the algorithm by von Neumann to retrieve the random bit string.

A.1.3 The combination

Jennewein et al. [20] describe two methods to obtain a random bit string, both using a continuous light. The first method is by using a 50:50 beam splitter and the second method polarizes the photon by 45° and a polarizing beam splitter is used. By polarizing the photon at 45°, the photon has a 50% chance to be in the horizontal polarization or in the vertical polarization. Both methods have two detectors, $D1$ and $D2$, which toggle a switch that gives the output. $D1$ corresponds to an output of 0, and $D2$ corresponds to an output of 1. If $D1$ detects a photon, it toggles the switch to 0. If $D2$ detects a photon, it toggles the switch to 1. If either $D1$ or $D2$ detect a photon and the switch is already at the position that it needs to toggle to, the switch will not be altered. The switch outputs its position (either 0 or 1), which produces the random bit string.

A.2 Radioactive decay

The time between radioactive decay of an element is another form of randomness based on quantum-theory. This was first investigated by Isida and Ikeda [19] by counting the number of output pulses by radioactivity of cobalt-60 in a constant

time interval. They state that the distribution of these amounts follow a Poisson distribution.

Hotbits [38] offers a service to provide random numbers based on radioactive decay. They detect the radioactive decay with a Geiger-Müller tube which is interfaced with a computer. They can produce random bytes at a speed of around 100 bytes per second. They do not elaborate on the source of the radioactive decay.

Although radioactive decay is a good source of physical randomness and well explored in the past, it is not directly applicable in an electronic device. This is the main reason why this source of randomness is not used often in combination with a cryptographic system on an integrated circuit.

A.3 Chaos-based True Random Number Generator

A chaos-based TRNG is a deterministic, but non-linear system. It is dependent on a initial condition, but one slight alteration in this initial condition causes large alterations in the output. This does mean that the system is predictable in the beginning up till a certain point. The divergence of the different trajectories in the system should be of exponential order, according to the Lyapunov exponent. However, Callegari et al. [11] proof that if no initial condition is known and the system is well designed, the output cannot be predictable. Chaos-based TRNG can be based on analog and digital phenomena.

A.3.1 Analog phenomena

One of the most well known analog chaos-based TRNG is the Chua circuit [23], which consists of one nonlinear element and a 3-segment piecewise-linear resistor. Because it is so simple, a lot of research is based on this Chua circuit. The downside of having these analog chaos-based TRNG, is the fact that everything is becoming digitized and that these analog chaos-based TRNG are thus becoming outdated.

A.3.2 Digital phenomena

Bernstein et al. [7] created a chaos-based TRNG with a digital phase-locked loop and elaborated on two important issues: the time one needs to wait to securely take a bit after one bit has been taken, given no initial conditions (4 to 8 iterations). The second issue is the waiting time one should have before sampling when the initial condition is known (around 20 iterations).

Zidan et al. [44] propose another fully digital, but differential, chaos based TRNG. They do however apply post-processing to remove some of the bias. Their system is however applicable to other more complicated analog RNG.

Another well known digital chaos-based TRNG is proposed by Stojanovski et al. [34], implemented on a VLSI.

Kamata et al. [21] proposed a chaos-based TRNG based on digital signal processing (e.g. a LAN). They implemented it on a FPGA and the algorithm they propose has perfect recovery characteristics of the transmitted data.

Appendix B

TestTool

The first target for this research was named TestTool (Spartan-6 FPGA). This area of research was still new, so some global measurements were taken first. Two (small) frequency sweeps were done afterwards.

B.1 Introduction

TestTool had 2 ROs implemented and running. One RO consisted of 51 inverters and one RO consisted of 61 inverters. This leads to operating frequencies of respectively 28 MHz and 24 MHz. For these experiments, 1 RO was measured at a time. The last inverter of the RO was mapped to an output-pin which the LeCroy was hooked up to. The location of the ROs is known, and the injection took place (roughly) on top of the RO that was measured.

B.2 Initial experiment

During the initial experiment some measurements without injection were taken to determine the exact operating frequency of the RO. As expected, the RO shifts in operating frequency, albeit that the shift is not much. Without injection, the RO operating at 24 MHz was found to be between $[23.956 - 24.414]$ MHz. Over 50% of the time the RO was operating at 24.109 MHz. The RO operating at 28 MHz had a varying frequency between $[27.924 - 28.381]$ MHz. Over 50% of the time this RO was operating at a frequency of 28.076 MHz. Four injections were done after the initial measurements. Two different frequencies were chosen (25 MHz and 26 MHz) and two different input powers (-2 dBm and -5 dBm). The RO consisting of 61 inverters (operating at 24 MHz) is monitored during these injections. During these injections it was found that the operating frequencies of the RO changes to a lower frequency outside of the previous window of operating frequencies, although the injected frequency is higher than the operating frequency. Another observation is the fact that the operating

frequency of the RO seems to be limited to a smaller window of frequencies the longer the injection takes place.

B.3 Frequency sweeps

Two frequency sweeps were performed. The frequency sweeps were from 25 MHz to 26 MHz with a step-size of 50 KHz. For one frequency sweep, the injection input power was set to -2 dBm. The other frequency sweep had an input power of -5 dBm. This frequency sweep looked at the operating frequency of the RO consisting of 51 inverters.

In the beginning of the frequency sweep, the operating frequency varies in the window of $[27.924 - 28.534]$ MHz. After 20 minutes it has a more stable operating frequency. For the RO operating at 28 MHz the operating frequency window shifted from $[27.924-28.381]$ MHz down to $[27.771-27.924]$ MHz during the frequency sweep at -5 dBm. During the frequency sweep at -2 dBm the operating frequency window shifted down to $[27.618-27.771]$ MHz. From these sweeps it is shown that injection does not only lower the operating frequency of the RO, but also narrows the window of operating frequencies the RO operates on. Narrowing this window could cause the ROs to behave more stable and can thus cause predictable output.

B.4 Visual random numbers

If a random number is failing the mono-bit test (counting the number of 1's and/or 0's), then it might be visual if the number is drawn. Common practice for drawing a random number is to draw a black square if the bit in the number is 1, or a white square if the bit is 0. Figure B.1 shows the output of a random number generator drawn this way. Figure B.2 shows visual representation of the random number produced by TestTool. Figure B.2 is the output of TestTool without injection. As can be clearly seen, TestTool would not pass the random number test suite.

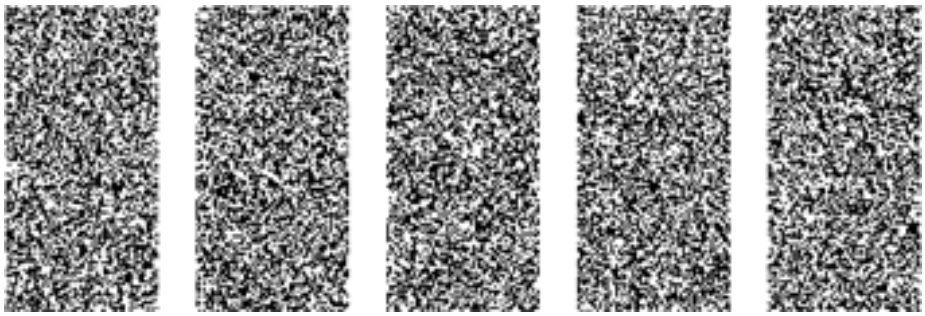


Figure B.1: Real random number generator output

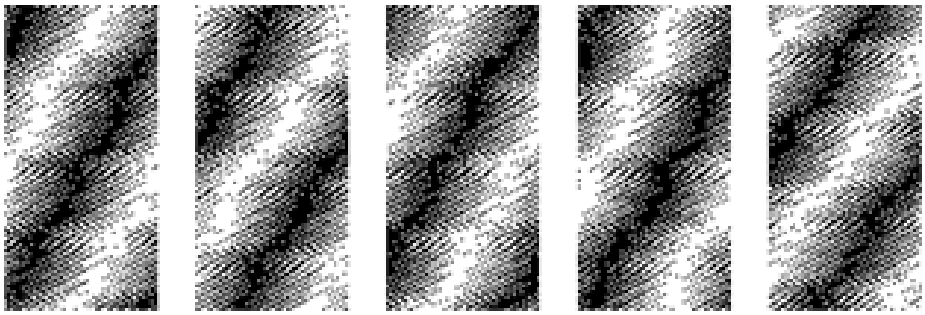


Figure B.2: TestTool random number generator output with 2 ROs of length 51 and 61

The reason for the inability to produce random numbers might be because of the sampling frequency. The sampling frequency of the implementation is higher than the operating frequency of the ROs, leading to the sampling of the same bit from the XOR-tree. Since the sampling frequency was harder to change (it was used for more parts in the design) than the amount of inverters, the latter was chosen. A new design was created with 2 ROs, both consisting of 5 inverters. Figure B.3 shows the random number produced by the new design in chronological order from the start of the injection up till 40 minutes. The resulting design does not produce random numbers, although the sampling frequency is less than the operating frequency. Research shows that the sampling frequency of ROs can change the random behavior of TRNGs based on ROs significantly. Although the design does not produce random numbers, an injection was done at 224 MHz on -5 dBm. Figure B.4 shows the visual representation of the random number. The first depicts the moment where the injection just started and an amplification of the pattern that was already present can be seen. Figure B.4 shows the visuals in a chronological order from 0 minutes to 30 minutes. Each point in time shows a bigger amplification of the pattern.

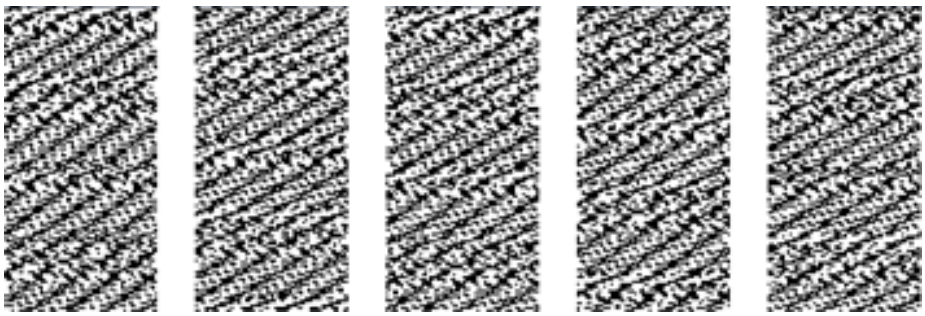


Figure B.3: Output without injection

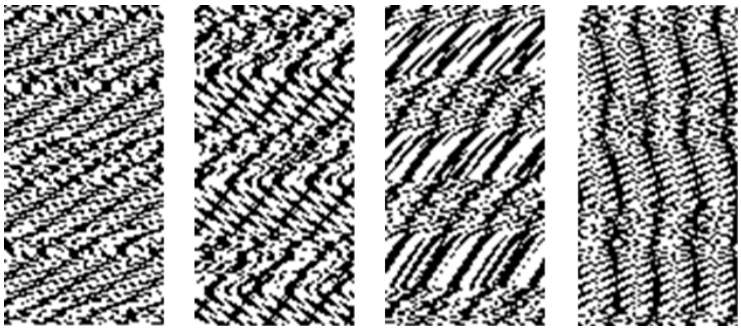


Figure B.4: Output during injection on 224 MHz on -5 dBm

B.5 Temperature

The research performed by Bayon et al had an almost instant result in their bias of the TRNG. The results presented here only occur over some time. A reason, other than our injection, might be because of the temperature the injection induces into the chip. The amplifier itself is giving off heat, which is blown on top and heating the surface of TestTool. Also the electric coupling of the EM-field is increasing the temperature in TestTool. An increased temperature also explains the drop of the operating frequencies of the ROs while injecting a higher frequency. To test the effect of temperature, an injection of 224 MHz on -5 dBm was done again in combination with the appliance of cooling spray to the chip. Figure B.5 shows the visual result of the output of TestTool during injection with cooling spray applied. As can be seen, the visual output is showing different patterns each time. Temperature affects the operating frequency of the ROs and therefore also has an effect on the output. However, the first random number from Figure B.5 does show a lot of resemblance to the last random number of Figure B.4. It therefore seems that both temperature and the injected EM frequency have an effect on the output of TestTool.

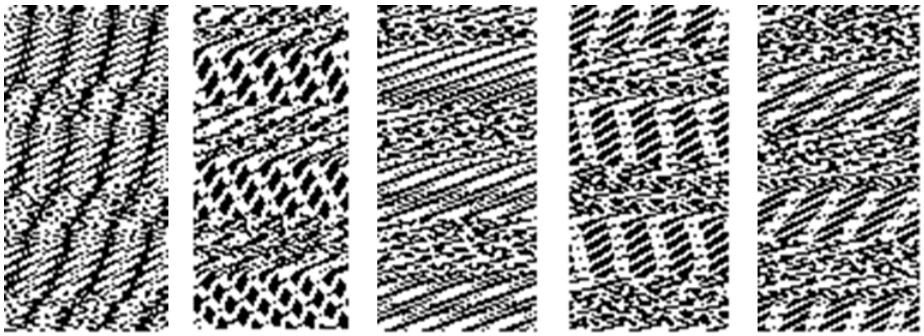


Figure B.5: Applying cooling spray during injection

B.6 Conclusion

The experiments on TestTool give useful insight how ROs react to the injection of an EM FI using harmonic emission. The injected frequency locks the ROs to a different frequency, but also the temperature induced has an effect on the operating frequency of the ROs. Injecting an EM harmonic signal also seems to make the RO operate on a smaller window of frequencies. The disadvantage of TestTool is an implementation problem that fails to produce random numbers. Therefore the effect of the injection on the output of random numbers cannot be quantified.

TestTool had issues producing random numbers. The main target of this research is an Actel Fusion. This FPGA was also used in the research performed by Bayon et al. The Actel Fusion FPGA is able to produce statistical random numbers according to random number test suites (NIST 800-22, Dieharder and AIS-31).

Appendix C

Frequency sweeps on 2 ROs

Figure C.1 shows the result of the frequency sweep for the north-west corner. The large peak at 258 MHz corresponds to the operating frequency of the second RO, the peak at 250 MHz is the operating frequency of the first RO and the peak at 254 is the mean of the two peaks. Figure C.2 shows the result of the frequency sweep performed on the north-east corner. The peaks at the positions mentioned for Figure C.1 are the same in this case.

Figure C.3 shows the result of the frequency sweep located on the south-west corner. Although the peak positions are the same here, it seems like the highest MI is achieved when injecting the frequency of the first RO in this case.

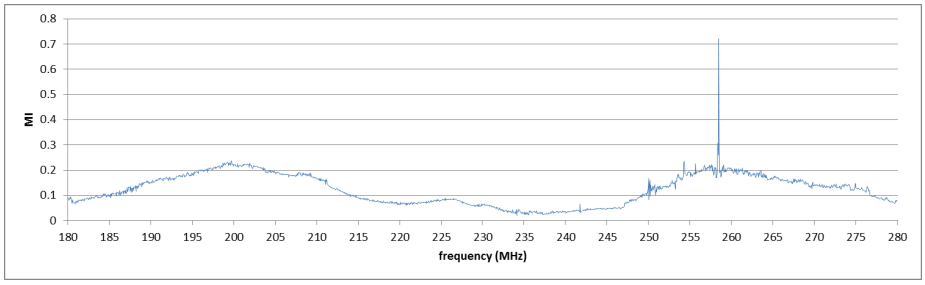


Figure C.1: Frequency sweep in the north-west corner

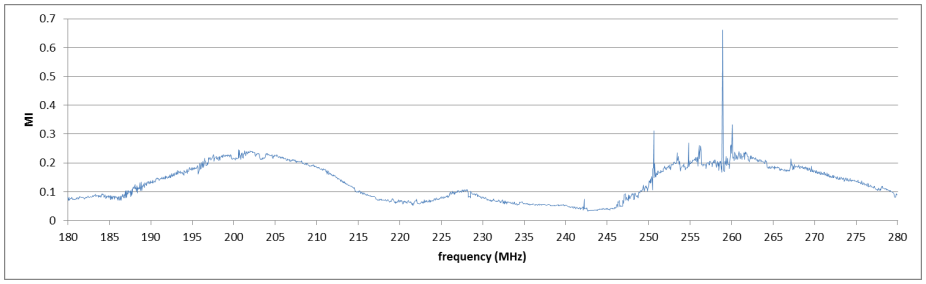


Figure C.2: Frequency sweep in the north-east corner

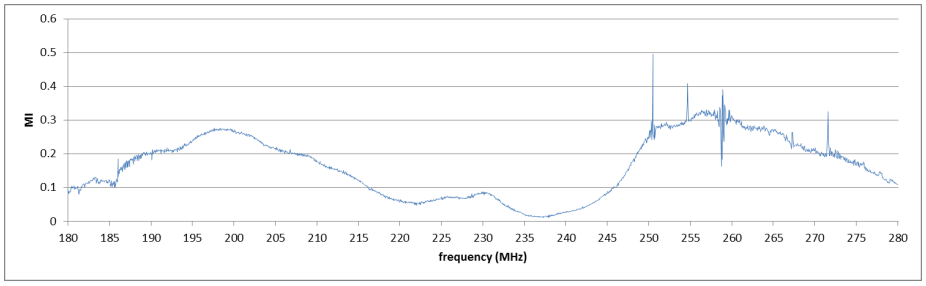


Figure C.3: Frequency sweep in the south-west corner

Appendix D

Experiments with a short probe

A frequency sweep was also performed on the design consisting of 5 ROs (see Chapter 5) The whole setup is the same as described before, except for the probe, which is a shorter and isolated probe. It is assumed that using this probe should give a more localized effect. However, it was found that the long probe also has a local effect on certain occasions. This appendix will elaborate on the results from the short probe.

Figure D.1 shows the frequency sweep performed and shows the values of the peak of the injection frequency divided by the peak of the operating frequency of the RO. As can be seen, the effect is not very large compared to injection with a long probe. The highest value is less than 2, while going over 80 with the long probe. The peak of the injection frequency was not visible in the FFT-spectrum on certain occasions, resulting in a value of 0.

The top 6 injection frequencies for both methods to find a good injection frequencies are shown in Table D.1. The first method gives us a window from 267 MHz to 270 Mhz, while the second method gives optimum injection frequencies from 254 MHz to 260 MHz. These are different optimum injection frequencies compared to the long probe, which is to be expected. A different length of the probe results in different optimal injection frequencies. However, both methods define a different range of optimum injection frequencies. For these experiments the MI between ROs was not checked because of a limited time window.

Unfortunately at the time of the experiment it was believed that a bad ratio would result in no good injection frequency. However, at the time of writing it is believed that the location of the injection was on an ineffective spot and that this short probe could still potentially lead to a biased TRNG output.

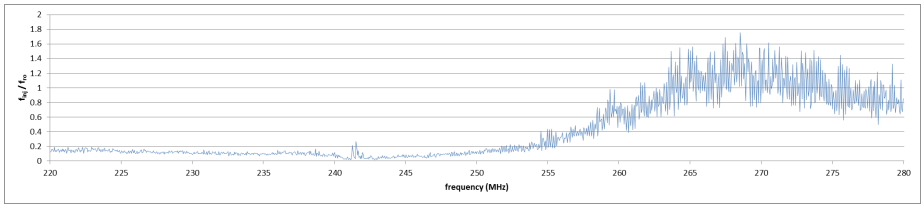


Figure D.1: Frequency sweep with the short isolated probe

Nr	$dB(f_{inj})/dB(f_{RO})$	$dB(f_{ROinj}) - dB(f_{RO})$
1	268.5 MHz	254.7 MHz
2	267.45 MHz	255.55 MHz
3	270.5 MHz	254.95 MHz
4	268.2 MHz	259.75 MHz
5	268.95 MHz	256.6 MHz
6	267.3 MHz	254.5 MHz

Table D.1: Top 6 injection frequencies with a short isolated probe

Appendix E

All results of different designs

This appendix shows all the results of every all the designs tested in Section 6.2. The first XY-plot is the average of 3 XY-scans with an injection frequency equal to the operating frequency of the first RO. The second XY-plot is the average of 3 XY-scans with an injection frequency equal to the operating frequency of the second RO and the third XY-plot is the average of 3 XY-scans with an injection equal to the mean of the operating frequencies of both ROs. All results are based on the MI. The colors are based on the minimum and maximum of every design, and therefore the coloring of different designs cannot be compared.

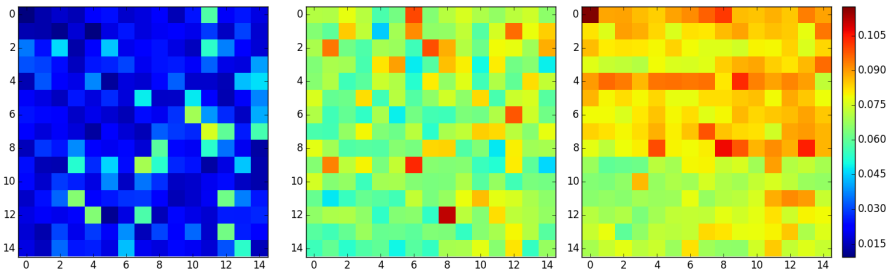


Figure E.1: Horizontal, with antenna, in-line, equal length

Figure E.1 shows the result of a design with horizontal ROs, both with antenna, in-line and of equal length. The mean of the operating frequencies looks like the best injection (although MI is only going up to 0.105) positioning the probe to the top part of the chip. Injecting the operating frequency of RO2 also seems to produce the same MI, with some maximum values next to the wire going to the XOR-gate. The spot (12,8) might be an influence on the bonding

wire connecting the signal of the RO to the output pin. Injecting the frequency of RO1 seems to have little to no effect.

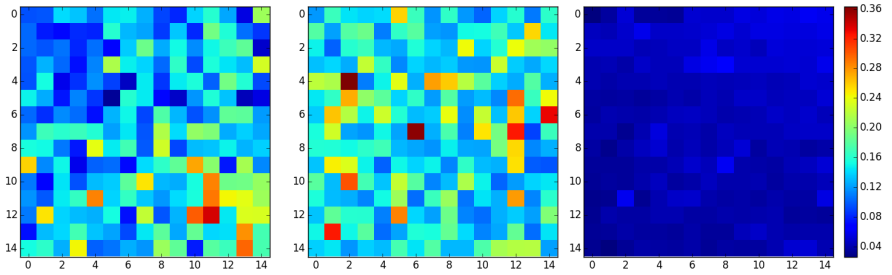


Figure E.2: Horizontal, with antenna, in-line, unequal length

Figure E.2 shows the result for horizontally unequal length ROs, with the first RO having an antenna and in-line. Injection of the operating frequency of RO1 has some high MI values on the bottom right corner. This might be an influence on the bonding wire connecting the signal of the RO to an output pin. Injection of the operating frequency of RO2 seems to influence the ROs. The coordinate (7,6) is next to the location of the ROs. There is also an orange and yellow square next to the wire going from the ROs to the XOR-gate. MI goes up to 0.36 when injecting the operating frequency of RO2, but when injecting the mean of the operating frequencies it shows no effect.

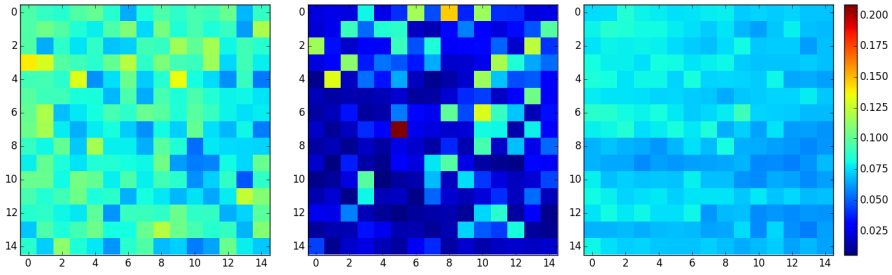


Figure E.3: Horizontal, with antenna, parallel, equal length

Figure E.3 shows the result for horizontal, parallel placed equal length ROs, both with an antenna. Injecting a frequency equal to the operating frequency of RO1 has a somewhat global good effect, except for the squares around (9,10). When injecting the operating frequency of RO2, a high MI is achieved next to the ROs again. Spots along the wire going to the XOR-gate also look like good injection locations.

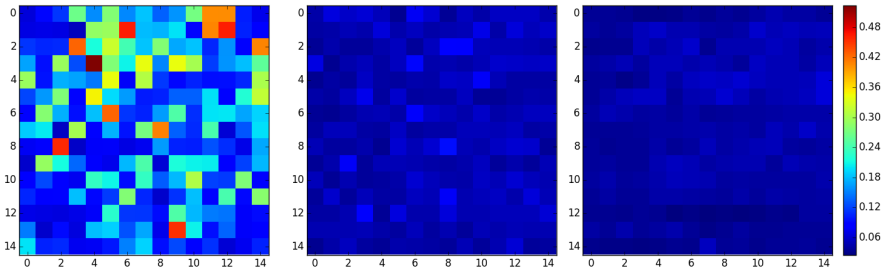


Figure E.4: Horizontal, with antenna, parallel, unequal length

Figure E.4 shows the result for a design that implements the ROs of unequal length horizontally in parallel, with an antenna for the first RO. Injection on the operating frequency of RO2 has no effect. Injecting on the mean of the operating frequencies has no effect. When injecting with the operating frequency of RO1, MI goes up to 0.5. Spots near the wire from the last element of the ROs to the XOR-gate seems to be the best point of injection.

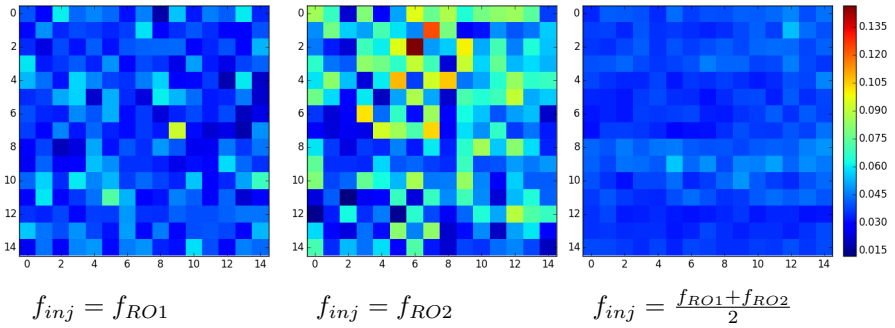


Figure E.5: Horizontal, without antenna, in-line, equal length

Figure E.5 shows the results for the design where horizontal ROs of equal length are placed horizontally and in-line. The only good injection spots are achieved when injecting the operating frequency of RO2 near the wire going to the XOR-gate. The MI does not go over 0.14 in these spots, which is not much.

Figure E.6 shows the results for a design that has horizontal ROs of unequal length, with an antenna for the first RO, while being placed in-line of each other. Injecting the operating frequency of RO1 has no effect. Injecting the operating frequency of RO2 has a mixed effect, with high MI on the bonding wires connecting the signal of the ROs to the output pins and on the wire going to the XOR-gate. The mean of the operating frequencies results in a global MI of on average 0.33.

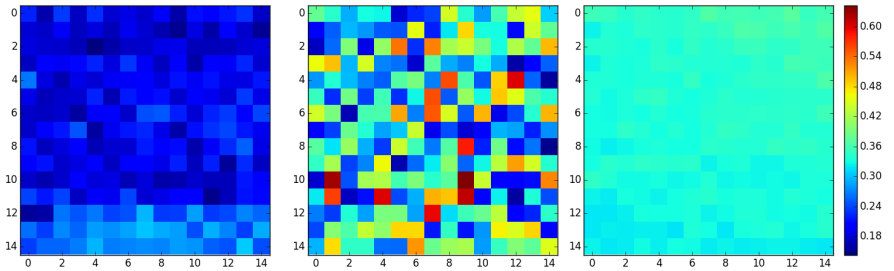


Figure E.6: Horizontal, without antenna, in-line, unequal length

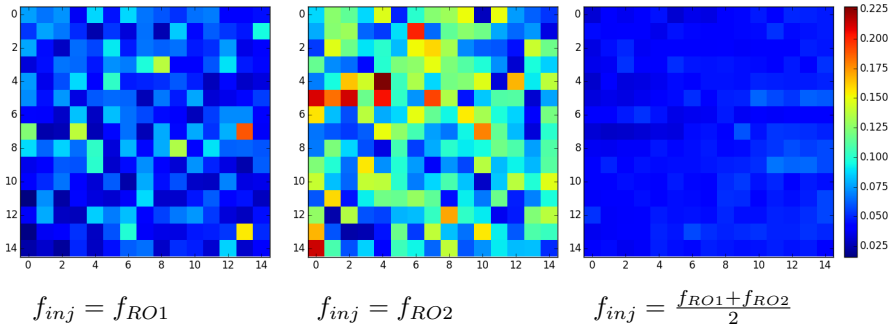
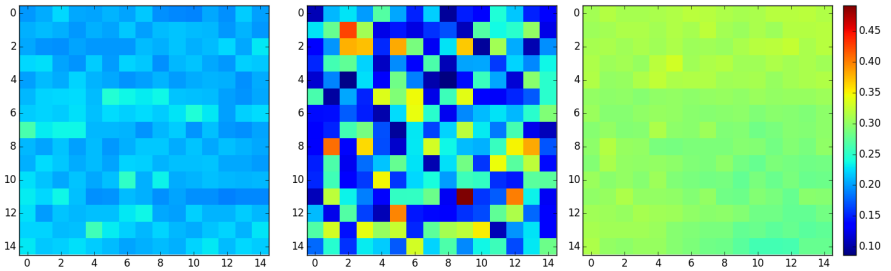


Figure E.7: Horizontal, without antenna, parallel, equal length

Figure E.7 shows the design with 2 horizontally ROs of equal length, without antenna and placed in parallel. Injecting the mean of the operating frequencies shows no effect. There are some good spots when injecting the operating frequency of RO1, but the highest MI is achieved at (7,13). No reason for the effectiveness of this spot can be given. Injecting the operating frequency of RO2 gives a high MI above the location of the ROs and next to the wire going to the XOR-gate.

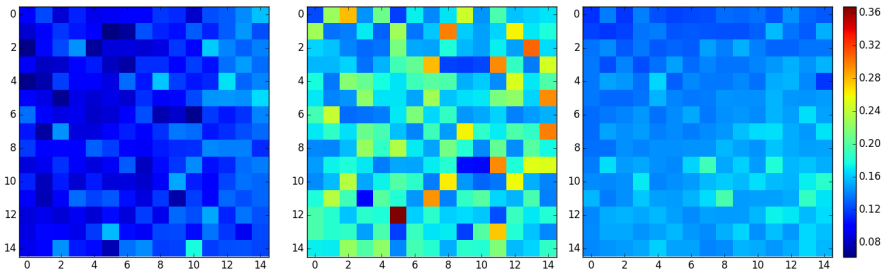
Figure E.8 shows the results for a design with horizontal ROs of unequal length with no antenna placed in parallel. Injection of the operating frequency of RO1 gives a decent global MI ranging from 0.2 to 0.3. Injecting the operating frequency of RO2 has a varying result. It seems to perform best when it is located outside of the programmable die. The wire going to the XOR-gate is also a good injection spot. Injecting the mean of the operating frequencies gives a global MI of 0.31.

Figure E.9 shows the result for a design with vertical ROs of equal length with an antenna placed inline. Injection of the operating frequency of RO1 seems to have no effect. Injection of the operating frequency of RO2 seems to



$f_{inj} = f_{RO1}$ $f_{inj} = f_{RO2}$ $f_{inj} = \frac{f_{RO1}+f_{RO2}}{2}$

Figure E.8: Horizontal, without antenna, parallel, unequal length



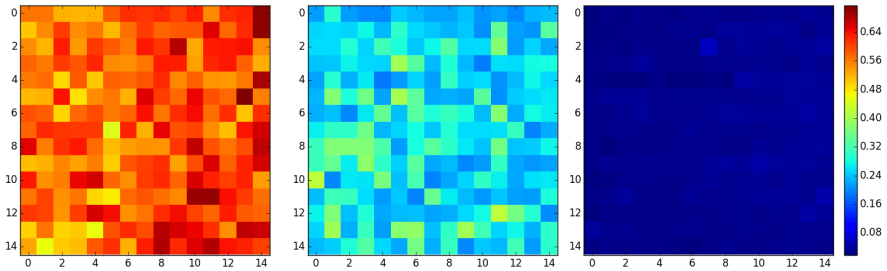
$f_{inj} = f_{RO1}$ $f_{inj} = f_{RO2}$ $f_{inj} = \frac{f_{RO1}+f_{RO2}}{2}$

Figure E.9: Vertical, with antenna, in-line, equal length

have good MI next to the wire going to the XOR-gate. It has the highest MI on (12,5), which might be an influence of the bonding wire and not necessarily on the ROs themselves. Injection of the mean of the operating frequencies seems to perform decent, with an MI ranging from 0.1 and 0.2.

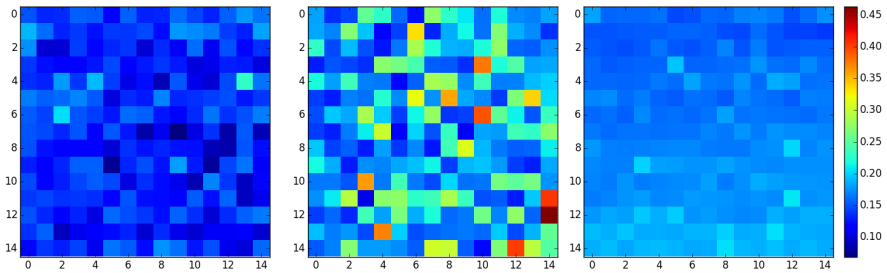
Figure E.10 shows the result for in-line placed vertical ROs of unequal length, with an antenna for the first RO. Injecting the frequency of the operating frequency of RO1 seems to produce a high MI of 0.6, which seems to be global. Injecting the mean of the operating frequencies seems to produce a global ineffective result. The injection of the operating frequency of RO2 also seems to produce a somewhat global result, varying in MI between 0.2 and 0.45.

Figure E.11 shows the result for a design with vertical equal length ROs, both with antenna placed in parallel. Injecting the operating frequency of RO1 shows an MI of 0.25 on location (4,13). No apparent reason for this can be given. Injection of the operating frequency of RO2 seems to achieve a high MI when injecting on the bonding wire and along the wire from the last element of the RO towards the XOR-gate. The mean of the operating frequencies seems to perform somewhat good, with an MI of 0.2.



$f_{inj} = f_{RO1}$ $f_{inj} = f_{RO2}$ $f_{inj} = \frac{f_{RO1}+f_{RO2}}{2}$

Figure E.10: Vertical, with antenna, in-line, unequal length



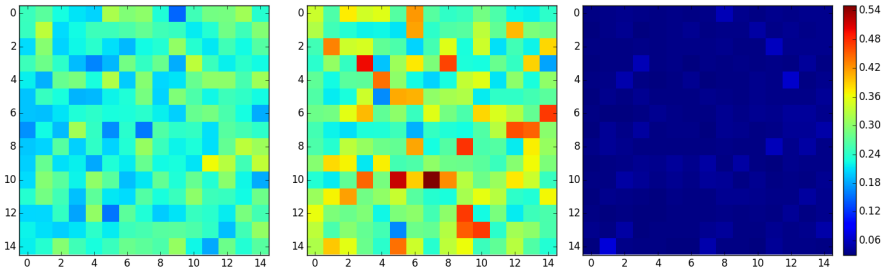
$f_{inj} = f_{RO1}$ $f_{inj} = f_{RO2}$ $f_{inj} = \frac{f_{RO1}+f_{RO2}}{2}$

Figure E.11: Vertical, with antenna, parallel, equal length

Figure E.12 shows the results for a design with unequal length ROs with an antenna for the first RO, placed vertical and parallel. Injecting the mean operating frequency results in a global low MI. Injecting the operating frequency of RO2 reveals good injection spots under the ROs and along the wire connecting to last element of the ROs to the XOR-gate. Injecting the operating frequency of RO1 seems to show a higher MI when located at the right side of the scan area.

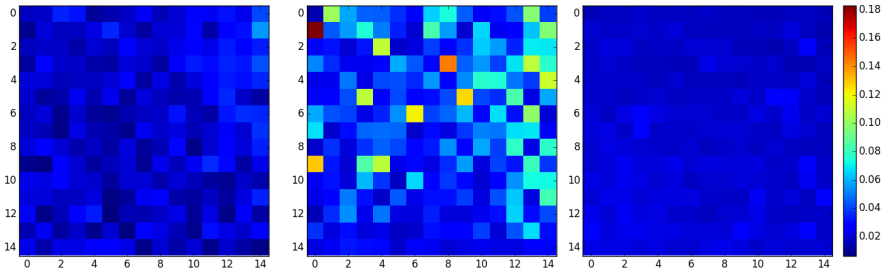
Figure E.13 shows that the operating frequency of RO1 and the mean of the operating frequencies are no good injection frequencies on a design that has vertical ROs of equal length with no antenna placed inline. Injection of the operating frequency of RO2 shows an MI of 0.14 near the wire connecting the third element to the XOR-gate. The spot (1,0) has the highest MI, though a reason for this cannot be given.

Figure E.14 shows the results for vertically placed ROs of unequal length without an antenna, placed in-line. Injecting the frequency of RO1 or the mean of the operating frequencies has no effect. Injecting the operating frequency of RO2 shows good spots to the left of the location of the ROs. This is also the



$f_{inj} = f_{RO1}$ $f_{inj} = f_{RO2}$ $f_{inj} = \frac{f_{RO1}+f_{RO2}}{2}$

Figure E.12: Vertical, with antenna, parallel, unequal length



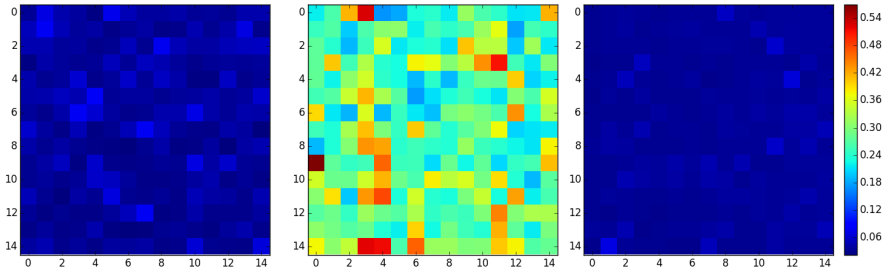
$f_{inj} = f_{RO1}$ $f_{inj} = f_{RO2}$ $f_{inj} = \frac{f_{RO1}+f_{RO2}}{2}$

Figure E.13: Vertical, without antenna, in-line, equal length

wire going to the bonding wire. Inside the programmable die of the chip seems like the most ineffective for this design, although it is producing an MI of 0.36 on average.

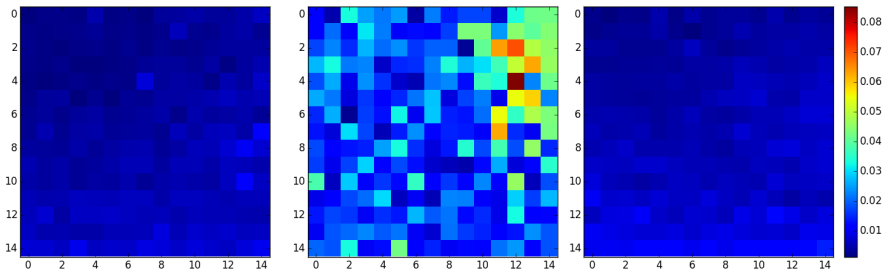
Figure E.15 shows the results for vertically placed ROs of equal lenght without antennas in parallel. MI is low for every chosen injection, but while injecting on the operating frequency of RO2 it seems the top right corner seems effective. It is unclear why, since it is outside of the programmable area of the die. This design seems to be very robust to this kind of attack.

Figure E.16 shows the results for ROs of unequal length, placed vertical in parallel, both without antenna. Injecting the operating frequency of RO1 seems to have no effect. Injecting the operating frequency of RO2 has effect next to the programmable die of the chip. Injecting the mean of the operating frequencies has a global good result, with an MI going over 0.6.



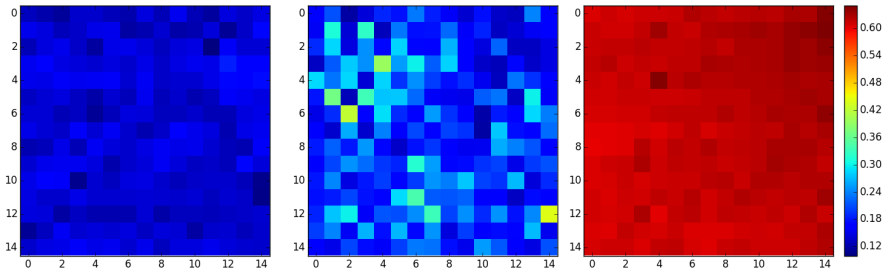
$f_{inj} = f_{RO1}$ $f_{inj} = f_{RO2}$ $f_{inj} = \frac{f_{RO1}+f_{RO2}}{2}$

Figure E.14: Vertical, without antenna, in-line, unequal length



$f_{inj} = f_{RO1}$ $f_{inj} = f_{RO2}$ $f_{inj} = \frac{f_{RO1}+f_{RO2}}{2}$

Figure E.15: Vertical, without antenna, parallel, equal length



$f_{inj} = f_{RO1}$ $f_{inj} = f_{RO2}$ $f_{inj} = \frac{f_{RO1}+f_{RO2}}{2}$

Figure E.16: Vertical, without antenna, parallel, unequal length

Bibliography

[1] R. Adler. A study of locking phenomena in oscillators. *Proceedings of the IEEE*, 61(10):1380–1385, Oct 1973.

[2] A. Alaeldine, T. Ordas, R. Perdriau, P. Maurine, M. Ramdani, L. Torres, and M. Drissi. Assessment of the immunity of unshielded multi-core integrated circuits to near-field injection. In *Electromagnetic Compatibility, 2009 20th International Zurich Symposium on*, pages 361–364. IEEE, 2009.

[3] D. Alberto, P. Maistri, and R. Leveugle. Forecasting the effects of electromagnetic fault injections on embedded cryptosystems. *Information Security Journal: A Global Perspective*, 22(5-6):237–243, 2013.

[4] P. Bayon, L. Bossuet, A. Aubert, and V. Fischer. Electromagnetic analysis on ring oscillator-based true random number generators. In *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*, pages 1954–1957. IEEE, 2013.

[5] P. Bayon, L. Bossuet, A. Aubert, V. Fischer, et al. Em radiation analysis on true random number generators: Frequency and localization retrieval method. In *Proceedings of the IEEE Asia-Pacific International Symposium and Exhibition on Electromagnetic Compatibility, APEMC 2013*, 2013.

[6] P. Bayon, L. Bossuet, A. Aubert, V. Fischer, F. Poucheret, B. Robisson, and P. Maurine. Contactless electromagnetic active attack on ring oscillator based true random number generator. In *Constructive Side-Channel Analysis and Secure Design*, pages 151–166. Springer, 2012.

[7] G. M. Bernstein and M. A. Lieberman. Secure random number generation using chaotic circuits. *Circuits and Systems, IEEE Transactions on*, 37(9):1157–1164, 1990.

[8] R.G. Brown. Dieharder test-suite. http://www.phy.duke.edu/ rgb/General/dieharder.php, 2014.

[9] S. Buchovecká. Analysis of a true random number generator. Master's thesis, Czech Technical University in Prague, Faculty of Information Technology, 2012.

[10] S. Buchovecká and J. Hlavac. Frequency injection attack on a random number generator. In *Design and Diagnostics of Electronic Circuits & Systems (DDECS), 2013 IEEE 16th International Symposium on*, pages 128–130. IEEE, 2013.

[11] S. Callegari, R. Rovatti, and G. Setti. Embeddable adc-based true random number generator for cryptographic applications exploiting nonlinear signal processing and chaos. *Signal Processing, IEEE Transactions on*, 53(2):793–805, 2005.

[12] A. Dehbaoui, J. Dutertre, B. Robisson, P. Orsatelli, P. Maurine, and A. Tria. Injection of transient faults using electromagnetic pulses-practical results on a cryptographic system-. *IACR Cryptology ePrint Archive*, 2012:123, 2012.

[13] J. F. Dynes, Z. L. Yuan, A. W. Sharpe, and A. J. Shields. A high speed, post-processing free, quantum random number generator. *Applied Physics Letters*, 93, 2008.

[14] J. D. Golic. New methods for digital generation and postprocessing of random data. *Computers, IEEE Transactions on*, 55(10):1217–1229, 2006.

[15] M. Hai-Qiang, W. Su-Mei, Z. Da, C. Jun-Tao, J. Ling-Ling, H. Yan-Xue, and W. Ling-An. A random number generator based on quantum entangled photon pairs. *Chinese Physics Letters*, 21(10):1961, 2004.

[16] Y. Hayashi, N. Homma, T. Sugawara, T. Mizuki, T. Aoki, and H. Sone. Non-invasive emi-based fault injection attack against cryptographic modules. In *Electromagnetic Compatibility (EMC), 2011 IEEE International Symposium on*, pages 763–767. IEEE, 2011.

[17] Y. Hayashi, N. Homma, T. Sugawara, T. Mizuki, T. Aoki, and H. Sone. Non-invasive trigger-free fault injection method based on intentional electromagnetic interference. *Proc. NIAT 2011*, 2011.

[18] W. T. Holman, J. A. Connelly, and A. B. Dowlatabadi. An integrated analog/digital random noise source. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 44(6):521–528, 1997.

[19] M. Isida and H. Ikeda. Random number generator. *Annals of the Institute of Statistical Mathematics*, 8(1):119–126, 1956.

[20] T. Jennewein, U. Achleitner, G. Weihs, H. Weinfurter, and A. Zeilinger. A fast and compact quantum random number generator. *Review of Scientific Instruments*, 71(4):1675–1680, 2000.

[21] H. Kamata, T. Endo, and Y. Ishida. Communication with chaos via dsp implementation. In *Circuits and Systems, 1997. ISCAS'97., Proceedings of 1997 IEEE International Symposium on*, volume 2, pages 1069–1072. IEEE, 1997.

[22] A. T. Markettos and S. W. Moore. The frequency injection attack on ring-oscillator-based true random number generators. In *Cryptographic Hardware and Embedded Systems-CHES 2009*, pages 317–331. Springer, 2009.

[23] T. Matsumoto. A chaotic attractor from chua's circuit. *Circuits and Systems, IEEE Transactions on*, 31(12):1055–1058, December 1984.

[24] B. Mesgarzadeh and A. Alvandpour. A study of injection locking in ring oscillators. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, volume 8, pages 5465–5468, 2005.

[25] K. Nohl, D. Evans, S. Starbug, and H. Plötz. Reverse-engineering a cryptographic RFID tag. In *Proceedings of the 17th Conference on Security Symposium*, SS'08, pages 185–193, Berkeley, CA, USA, 2008. USENIX Association.

[26] National Institute of Standards and Technology. NIST SP 800-22 test-suite. http://csrc.nist.gov/groups/ST/toolkit/rng/index.html, December 2008.

[27] F. Poucheret, K. Tobich, M. Lisarty, L. Chusseau, B. Robisson, and P. Maurine. Local and direct em injection of power into cmos integrated circuits. In *Fault Diagnosis and Tolerance in Cryptography (FDTC), 2011 Workshop on*, pages 100–104. IEEE, 2011.

[28] J. Schmidt and M. Hutter. Optical and EM fault-attacks on crt-based RSA: Concrete results. 2007.

[29] W. Schottky. Spontaneous current fluctuations in various conductors. *Ann. Physik*, 57:541–567, 1918.

[30] M. Schutten, S. Prabhakaran, D. Karipides, J. Nasadoski, and R. Thomas. High frequency emi filter parasitic characterization. In *Vehicle Power and Propulsion Conference (VPPC), 2011 IEEE*, pages 1–8, Sept 2011.

[31] M. Šimka and P. Komenského. Active non-invasive attack on true random number generator. In *6th PhD Student Conference and Scientific and Technical Competition of Students of FEI TU Košice, Košice, Slovakia*, pages 129–130, 2006.

[32] M. Soucarros, C. Canovas-Dumas, J. Clédière, P. Elbaz-Vincent, and D. Réal. Influence of the temperature on true random number generators. In *Hardware-Oriented Security and Trust (HOST), 2011 IEEE International Symposium on*, pages 24–27. IEEE, 2011.

[33] A. Stefanov, N. Gisin, O. Guinnard, L. Guinnard, and H. Zbinden. Optical quantum random number generator. *Journal of Modern Optics*, 47(4):595–598, 2000.

[34] T. Stojanovski, J. Pihl, and L. Kocarev. Chaos-based random number generators. part ii: practical realization. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, 48(3):382–385, 2001.

[35] B. Sunar, W. J. Martin, and D. R. Stinson. A provably secure true random number generator with built-in tolerance to active attacks. *Computers, IEEE Transactions on*, 56(1):109–119, 2007.

[36] G. Taylor and G. Cox. Behind intels new random-number generator. http://spectrum.ieee.org/computing/hardware/behind-intels-new-randomnumber-generator, August 2011.

[37] R. Velegalati, R. Van Spyk, and J. van Woudenberg. Electro magnetic fault injection in practice. 2013.

[38] J. Walker. Hotbits: Genuine random numbers, generated by radioactive decay. https://www.fourmilab.ch/hotbits/, May 1996.

[39] K. Wold and S. Petrovic. Security properties of oscillator rings in true random number generators. In *Design and Diagnostics of Electronic Circuits & Systems (DDECS), 2012 IEEE 15th International Symposium on*, pages 145–150. IEEE, 2012.

[40] K. Wold and C. H. Tan. Analysis and enhancement of random number generator in fpga based on oscillator rings. *International Journal of Reconfigurable Computing*, 2009:4, 2009.

[41] P. Xu, Y. L. Wong, T. K. Horiuchi, and P. A. Abshire. Compact floating-gate true random number generator. *Electronics Letters*, 42(23):1346–1347, 2006.

[42] S. Yoo, B. Sunar, D. Karakoyunlu, and B. Birand. A robust and practical random number generator, 2007.

[43] Z. L. Yuan, B. E. Kardynal, A. W. Sharpe, and A. J. Shields. High speed single photon detection in the near infrared. *Applied Physics letters*, 2007.

[44] M. A. Zidan, A. G. Radwan, and K. N. Salama. Random number generation based on digital differential chaos. In *Circuits and Systems (MWSCAS), 2011 IEEE 54th International Midwest Symposium on*, pages 1–4. IEEE, 2011.

[45] L. Zussa, A. Dehbaoui, K. Tobich, J. Dutertre, P. Maurine, L. Guillaume-Sage, J. Clediere, and A. Tria. Efficiency of a glitch detector against electromagnetic fault injection. In *Proceedings of the conference on Design, Automation & Test in Europe*, page 203. European Design and Automation Association, 2014.