Experimental Study of Locking Phenomena on Oscillating Rings Implemented in Logic Devices

Ugo Mureddu, Nathalie Bochard, Lilian Bossuet, Viktor Fischer

Experimental Study of Locking Phenomena on Oscillating Rings Implemented in Logic Devices

Ugo Mureddu, Nathalie Bochard, Lilian Bossuet, Viktor Fischer

***Abstract*—Oscillating rings are widely used in CMOS logic devices because they are easy to integrate, require low area and low power. Their main disadvantage is that they tend to lock to each other and/or to an external periodic signal. This locking phenomenon can render a system based on a freely running oscillator non-functional. A detailed study of the causes of the phenomenon and how to avoid it, is therefore of paramount importance. In this paper, we conduct a detailed examination of the locking phenomenon using the most commonly used rings: ring oscillators, transient effect ring oscillators and self-timed rings. We then analyze the consequences of locking on different use cases based on oscillating rings and provide design recommendations to minimize its impact. Our results could help designers better anticipate locking phenomenon in their future designs. To ensure reproducibility of the results, the VHDL code of all the experiments is available and can be downloaded from a dedicated web page.**

***Index Terms*—Free running oscillators, ring oscillator, transient effect ring oscillator, self-timed ring oscillator, locking phenomenon**

I. Introduction

Electronic oscillators are key elements in many data processing applications. They are used in communication systems for radio and television signal modulation and demodulation or channel selection. They are also used in most digital circuits as a time reference for synchronizing operations or for serial data communication [1]. PLL based frequency synthesis and clock signal generation are also based on oscillators [2], [3]. In data security applications, oscillators serve as source of entropy for true random number generators (TRNGs) [4], [5] or physical unclonable functions (PUFs) [6], [7].

Ideally, electronic oscillator produces a perfect time reference (*i.e.* a periodic signal, often with a sine waveform or a square waveform). In practice, all types of oscillators are affected by perturbations and noises, and their output is not perfectly periodic. Numerous types of oscillator circuitries are available, but the principle of operation, the frequency stability and the robustness against variations in environmental conditions like supply voltage, temperature and electromagnetic interference are specific to each. The most commonly used are harmonic oscillators (*i.e.* RC or LC oscillators), quartz oscillators and relaxation oscillators (*i.e.* Pearson-Anson oscillators, comparator-based oscillators or CMOS oscillating rings) [8], [9].

U.Mureddu, N.Bochard, L.Bossuet and V.Fischer are with the Secure Embedded System and Hardware Architecture group in the computer sciences Department of the Hubert Curien Laboratory, University of Lyon, UJM-Saint-Etienne, CNRS, UMR 5516, F-42023, Saint-Etienne, France (email:ugo.mureddu@univ-st-etienne.fr)

Among them, CMOS oscillating rings are particularly interesting in digital integrated circuits (IC) since they are easy to implement using simple logic gates, need low area and low power and are consequently low cost. For all these reasons, they are widely used in ICs [10]. Their main known weakness is related to their limited immunity to perturbations [11], [12], [13].

Another phenomenon, that has not been sufficiently studied and taken into account, is their ability to lock onto a signal with a frequency close to their natural oscillation frequency or its harmonics. Interestingly, the interaction between two oscillatory systems operating at close frequencies and spatially close to each other has been known for centuries. In their paper, Mesgarzadeh and Alvandpour [14] mentioned that the locking phenomenon was observed for the first time in the 17^{th} century.

Although certain applications like frequency dividers take advantage of it [15], in most cases locking is something the designers want to avoid. Indeed, as the consequence of a malicious attack or of a coincidence caused by two signals oscillating at close frequencies, the locking phenomenon can render a system based on a freely running oscillator non-functional.

Imagine, for example, in a serial data communication, a time reference generated using an oscillating ring set at a frequency f_0 and perturbed by a signal of frequency f_0' close to f_0 in such a way that the first one is locked to the second one, the serial communication will be erroneous due to the shift of the reference frequency f_0 to f_0'.

Locking of oscillating rings used in data security applications could be even more prejudicial since it would compromise the confidential key generated using the rings in a TRNG. Indeed, the generated key would be partially or even fully deterministic and hence predictable [16].

In this paper, we conduct a detailed analysis of the locking phenomenon using the main types of oscillating rings based on logic gates: ring oscillators (ROs), transient effect ring oscillators (TEROs), and self-timed rings (STRs).

After presenting the general principles and similarities between the oscillating rings, we demonstrate that the locking phenomenon also affects TEROs and STRs and not only ROs, as assumed up to now [7].

Moreover, while the locking effect has already been observed on commonly used ROs, an evaluation of its impact and design recommendations for reducing it are still lacking.

Finally, we discuss the consequences of the locking phenomenon for different architectures and topologies of oscil-

lating rings implemented in logic devices and provide design recommendations to minimize its impact.

We investigated the locking phenomenon on field programmable gate arrays (FPGAs) for two reasons: 1) many architectures and topologies can be studied by reconfiguring the FPGA device; 2) the frequencies of oscillations in rings implemented in FPGAs are lower than in ASICs and can be observed even outside the device by using a low voltage differential signaling (LVDS) interface, together with differential oscilloscope probes. The locking phenomenon was studied on the three main FPGAs manufacturers: Xilinx Spartan 6 and Intel Cyclone V representing SRAM based FPGA devices and Microsemi SmartFusion 2 representing FLASH based FPGA devices.

The rest of the paper is organized as follows. In the following section, we summarize all publications related to the locking of oscillating rings implemented in logic devices. In Section III, we present the three main structures of oscillating rings studied in the paper. In Section IV, we provide information on the background of the locking phenomenon, experimental setups and proofs of locking used in the rest of the paper. In sections V, VI, and VII, we present the experimental results we obtained using different types of oscillating rings in different FPGA devices. In Section VIII, we compare and discuss our results and provide design recommendations to reduce locking phenomenon in oscillating rings as much as possible. In Section IX, we investigate the effect of signal routing on the locking of rings. Finally, in section X, we demonstrate the impacts of locking on concrete use cases based on oscillating rings implemented in logic devices.

All VHDL sources are available on git-lab to ensure repeatability[1].

II. Related work

Currently, only a few papers deal with the locking effect in oscillating rings based on logic gates, and those do only concern ROs.

The first category of papers represents frequency divider applications. The idea is to lock the RO to a periodic signal at a frequency that corresponds to the N^{th} harmonic of the ring oscillations. Among the most popular, we cite a study by Mirzaei *et al.* [17]. Based on the study of the input lock-in range of an RO, the authors show that injection at multiple stages of the RO improves the lock-in range. They demonstrate this effect on a divider by two and by six prototype. Other studies related to this topic are available in [15], [18].

The second category of papers targets security applications. For example, Bochard *et al.* [19] and Bernard *et al.* [20] noticed that the locking phenomenon negatively affects both RO-based TRNGs and RO-based PUFs.

Then, Marketos *et al.* [21] demonstrated that ROs can easily lock onto an external periodic signal injected into the ring via the power supply. In their paper, the authors showed that this kind of attack can significantly reduce the entropy rate at the output of the RO-based TRNG and make generated numbers manipulable. They provide a practical illustration of the attack on an EMV payment card.

Finally, Bayon *et al.* [16] present a contact-less attack using electromagnetic interference on RO based TRNGs. The authors show that ROs can lock onto a sufficiently strong electromagnetic field emitted near the device.

In all these papers, the authors showed that ROs are vulnerable to manipulation, but did not study their causes and did not analyze how these manipulations can be avoided or at least rendered more difficult.

III. Introduction to oscillating rings

As explained in Section I, the study focuses on oscillating rings built using logic gates. All the rings studied are composed of two basic elements: M activation gates used to trigger oscillations and N delay gates used to tune the oscillation frequency. Ring elements are connected in series and the output of the last element is looped back to the first to form a ring. The number of activation and delay gates depends on the type of ring and its expected behavior.

A rising edge at the input of the activation gate(s) triggers oscillations. Depending on the type of ring, one or more rising and/or falling edges can propagate across the ring at the same time. These edges, denoted e, are electrical events. If no event collision occurs, they can propagate permanently, or temporarily, when one edge catches the previous one, causing a collision, which stops oscillations.

The period of ring oscillations is double the sum of the delays of individual stages divided by the number of events e in the loop:

$$T = \frac{2 \times (M \times d_m + N \times d_n)}{e} \tag{1}$$

where d_m represents the mean delay of the activation gates and d_n the mean delay of the delay gates, M and N represent the number of activation and delay gates, respectively.

A. Ring oscillator

An RO is a single-event oscillating ring. It is composed of an odd number N of inverting gates (delay gates) and one AND gate used as an activation gate ($M = 1$). Figure 1 shows the architecture of a RO.

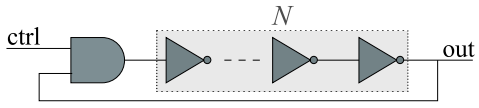


Fig. 1: Architecture of a ring oscillator (RO)

After the control signal *ctrl* moves from a logical low level to a logical high level, oscillations start. At any time, only one electrical event (rising or falling signal edge) is propagating across the ring ($e = 1$) – after crossing the ring, the rising edge is transformed to the falling edge and vice versa. Thus, the oscillation frequency equals:

$$f = \frac{1}{2 \times (d_{AND} + N \times d_{INV})} \tag{2}$$

where d_{AND} is the delay of the AND gate and d_{INV} is the mean delay of the inverters.

[1] https://gitlab.univ-st-etienne.fr/ugo.mureddu/locking-osicllating-cells.git

B. Transient effect ring oscillator

The TERO is a multi-event oscillating ring with event collisions. It has two states: one transient oscillatory state and one non-oscillating, stable state. As depicted in Fig. 2, the ring is composed of two chains of N inverters representing delay gates and two AND gates used as activation gates ($M = 2$). The TERO thus corresponds to a specific configuration of an RS latch [22] characterized by extended delays in the two latch branches.

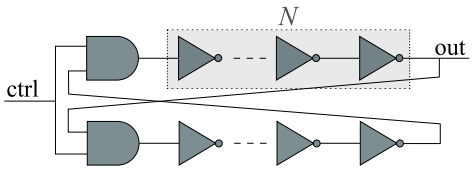


Fig. 2: Architecture of a transition effect ring oscillator (TERO)

When the control signal, denoted *ctrl* in Figure 2, goes high, two electrical events start to propagate across the ring ($e = 2$). Due to mismatches between the CMOS transistors composing the ring, caused by variations in the manufacturing process, one event is faster than the other. Consequently, while the output oscillation frequency does not change, the duty cycle moves towards 0% or 100% until the oscillations stop (see Fig 3).

Fig. 3: TERO behavior signal (*out*) after its stimulation using the control signal (*ctrl*)

Oscillation frequency of the TERO equals:

$$f = \frac{1}{(2 \times d_{AND} + 2 \times N \times d_{INV})} \tag{3}$$

where d_{AND} is the mean delay of the AND gates and d_{INV} is the mean delay of the inverters. TERO behavior is detailed in [23] and [24].

C. Self-timed ring oscillator

A self-timed ring (STR) is a multi-event oscillating ring without event collisions (see Fig. 4). Each stage of the STR (the STR cell) consists of a two-input Muller gate [25] and an inverter. Each STR cell initially serves as an activation gate and subsequently as a delay element. The neighboring STR cells communicate between themselves using a two-phase handshake protocol.

Several events can propagate without colliding thanks to this handshake protocol. The ring is initialized with e electrical events. An event present in a ring cell will move to the next cell if, and only if, the next cell is empty (i.e. if it does not already contain an event). For this reason, the STR needs at least one empty cell to oscillate. Consequently, up to $N - 1$ events can

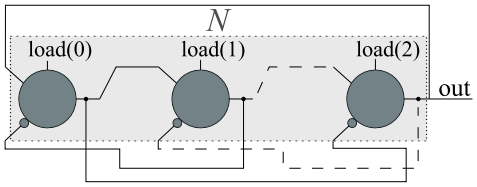


Fig. 4: Architecture of a self-timed ring oscillator (STR)

move across the loop consisting of N cells. Independently of their initial positions and thanks to analog mechanisms inferred in the ring, events end up either in a cluster that propagates in the ring (in a burst mode), or spread out around the ring and propagate with constant temporal spacing (in an evenly-spaced mode).

If there are more events than empty cells in the ring ($e > (N/2)$), the oscillation frequency is limited by the number of empty stages.

The oscillation frequency of the STR is expressed as:

$$f = \frac{a}{(N \times d_{MULLER})} \tag{4}$$

where a is the number of events (when $e < (N/2)$) or the number of empty STR cells (when $e > (N/2)$), and d_{MULLER} is the mean delay of STR cells. The STR behavior is detailed and its frequency modeled in [26].

IV. Locking phenomena in oscillators

A. Theoretical background

Like any oscillating system (*e.g.* Huygens pendulum clocks), electronic oscillators can lock to each other [27]. If a signal featuring a frequency close to the natural frequency of the oscillator or to its harmonics is somehow injected into the oscillator, the oscillator ceases to run freely and is forced to oscillate at another frequency, *i.e.* the frequency of the injected signal or its harmonics.

In three consecutive years, from 1945 to 1947, Tucker [28], Adler [29] and Huntoon *et al.* [30] studied the conditions that controlled locking of electronic oscillators. Tucker showed that suppression of free oscillations and establishment of forced oscillations in a triode oscillator depends on two conditions: the power (or voltage) of the perturbation signal must exceed a certain value and the oscillation frequency of the perturbation signal must be close enough to the natural frequency of the oscillator. Adler completed this study by deriving the condition for synchronization:

$$\frac{V_{pert}}{V_{osc}} > 2Q \left| \frac{f_{pert} - f_{osc}}{f_{osc}} \right|, \tag{5}$$

where V_{pert} is the output amplitude of the perturbation signal, V_{osc} the output amplitude of the free oscillator, Q the quality factor of the oscillator, f_{osc} the output natural frequency of the free running oscillator and f_{pert} the output frequency of the perturbation signal.

Consequently, the closer the output frequency of the perturbation signal is to the output frequency of the free-running signal, the lower is the minimum injection strength for locking. Then, Huntoon *et al.* generalized Adler's equation to other

types of oscillator. Later, Mesgarzadeh and Alvandpour proved that Adler's conditions also apply on CMOS ring oscillators [14]. We evaluated the impact of locking on CMOS oscillating rings with the following experimental setup.

B. Experimental setup

The locking phenomenon on CMOS oscillating rings was studied using the test bench shown in Fig. 5. This test bench is composed of:

- **An FPGA based hardware platform** – a set of evaluation boards (Evariste) consisting of a motherboard, which ensures communication with the host PC and several daughter boards featuring different types of FPGAs. Thanks to this multi FPGA support, we were able to evaluate the locking phenomenon on multiple FPGA families using the same hardware platform. More details about the Evariste hardware platform are available in [31].
- **A function generator**, Agilent Technologies 81160A generating square signals up to 330 MHz and sinusoidal signals up to 500 MHz. The function generator produced the perturbation signal and triggered oscillations in the rings studied here.
- **An oscilloscope**, LeCroy WaveRunner 640Zi with a frequency range up to 4 GHz and recording rate of 40 Giga-samples per second. The oscilloscope recorded the activation signal, the CMOS oscillating ring signal and the perturbation signal.
- **Differential probes**, LeCroy WL-PBus to transfer high frequency differential signals from the FPGA to the oscilloscope.
- **A computer** running scripts to control the oscilloscope and the function generator, and to configure the FPGAs.

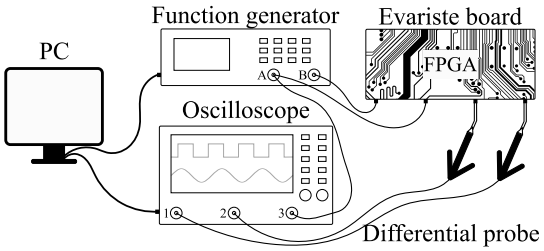


Fig. 5: Experimental setup

Output A of the function generator delivered the low frequency control signal used to activate oscillations in the ring and to trigger data acquisition in the oscilloscope. Output B delivered the perturbation signal at a frequency close to the natural frequency of the studied oscillating ring.

To mimic interactions between the ring and the surrounding logic, the perturbation signal was not injected electrically into the oscillating ring, but passed across a delay line placed near the ring. To maintain the strongest possible interactions between the ring and the delay line, all their neighboring elements were interleaved (see Fig. 6).

Outputs of the studied ring and the perturbation signal were observed with the oscilloscope using differential probes. Experiments were repeated with different frequencies of the perturbation signal. Algorithm 1 shows the pseudo code of the script written in Python.

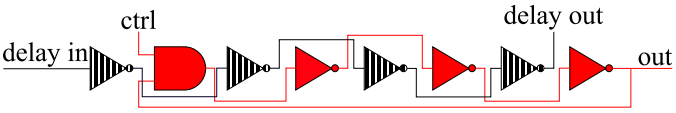


Fig. 6: Placement of the ring and the delay line in FPGA

Algorithm 1 Python script controlling the test bench

```
Program FPGA
Set activation signal
for f_pert = f_init to f_final do
  if rising edge of ctrl then
    Launch data acquisition
  end if
end for
```

C. Locking detection

Let the frequency of the ring output signal be f_{osc} and that of the ring output signal phase φ_{osc}. When locked to a perturbation signal, the ring deviates from its natural working conditions. Several electrical parameters can be measured to detect the locking phenomenon:

*1) **Frequency difference**:* in presence of a perturbation signal, the oscillator can lock its output signal frequency to that of the perturbation signal or to its harmonics. Once locked, the mean frequency difference between them, denoted $\overline{\Delta f}$, is 0. In the rest of the paper, $\overline{\Delta f}$ equals $\left|\overline{f_{pert} - f_{osc}}\right|$ (in Hz).

*2) **Period standard deviation**:* as explained in Sec. I, a CMOS oscillating ring is sensitive to noise and its output signal period (or frequency) is therefore not completely stable. Once locked, the output signal period of the ring is more stable than when it is free running. Consequently, the standard deviation of its period, denoted $\sigma_{T_{osc}}$ (in sec.), becomes less significant.

*3) **Phase shift**:* when the oscillator output is locked to the perturbation signal, the two signals do not necessarily have the same phase, but the phase shift between the two remains constant. The mean phase shift, denoted $\overline{\Delta_\varphi}$, equals $|\overline{\varphi_{pert} - \varphi_{osc}}|$, where φ_{pert} is the phase of the perturbation signal (given in °).

*4) **Phase shift standard deviation**:* in the same way, two independent signals have by definition a uniform cumulative distribution function of the phase shift, which corresponds to the standard deviation of the phase shift of $28,86\%$ or $104\,°$. This standard deviation, denoted σ_{Δ_φ}, tends to 0 when locked.

*5) **Additional parameter for TERO – number of oscillations**:* in the particular case of TERO, which in normal conditions should oscillate only temporarily, oscillations do not stop in the presence of locking. This state is quite easy to detect.

Designers will prefer one of above described metrics depending on what they want to detect (*e.g.* a transitional state between unlock and lock, or the beginning of locking) or depending on what kind of measurement they want to

implement (*e.g.* an external or embedded measurement in software or hardware, etc.).

Any system that is sensitive to the locking phenomenon has two distinct ranges of locking: the lock-in range and the capture range. The lock-in range is defined as the range of frequencies over which the locked system follows the changes in the f_{pert}. Capture range is the frequency range in which the unlocked system locks to the external signal. The capture range is always smaller than the lock-in range.

It was not possible to distinguish the difference between the lock-in range and the capture range in any of our experiments. For this reason, only the lock-in range is considered in the rest of the paper.

V. Locking effect in ring oscillators

A. RO frequency perturbation using a delay line

The experimental setup from Section IV-B was applied to a $N = 3$ RO implemented in a Xilinx Spartan 6 FPGA. Without any perturbations the natural oscillation frequency f_{osc} was 296 MHz.

Figure 7 shows the results of the four methods of locking detection described above.

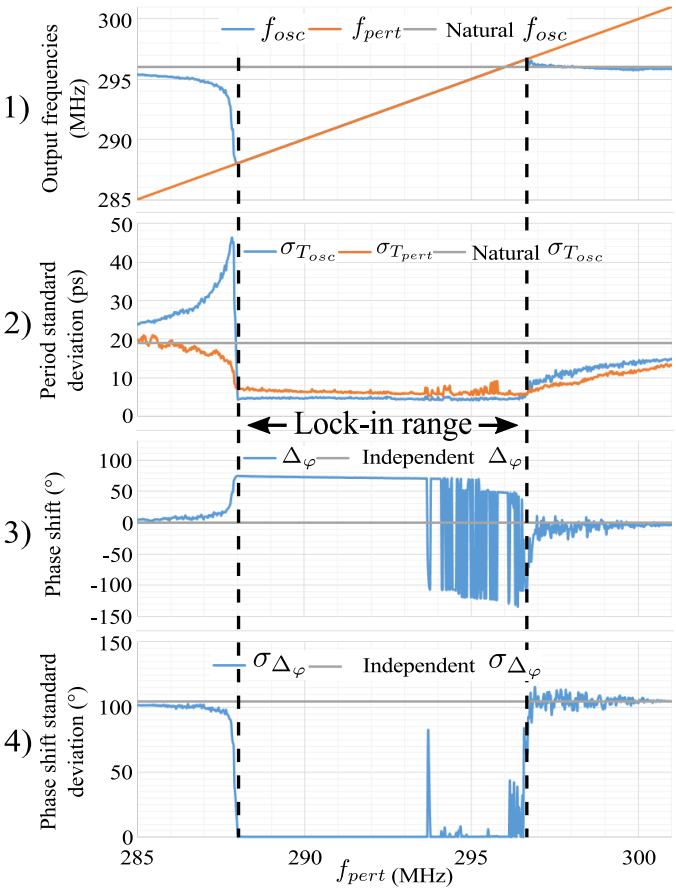


Fig. 7: Four locking detection methods demonstrated on an $N = 3$ RO implemented in the Xilinx Spartan 6 FPGA

1) $\overline{\Delta f}$: The topmost part of Fig. 7 depicts the evolution of the ring output signal frequency (f_{osc}) depending on the signal frequency of the perturbation signal (f_{pert}). The f_{pert} varied from 285 MHz to 301 MHz with steps of 0.025 MHz. At f_{pert} equal 285 MHz, the RO started to be perturbed and deviated slightly from its natural f_{osc}. Starting at 288 MHz the RO became locked to the perturbation signal and stayed locked up to 296.5 MHz. During the lock-in period, $\overline{\Delta f}$ was 0. The natural f_{osc} is also depicted to highlight the moment when the RO started to be perturbed.

2) $\sigma_{T_{osc}}$: The second part of Fig. 7 shows $\sigma_{T_{osc}}$ and $\sigma_{T_{pert}}$ depending on f_{pert}. Both $\sigma_{T_{osc}}$ and $\sigma_{T_{pert}}$ were measured using the oscilloscope. If the ring was not perturbed, the mean $\sigma_{T_{osc}}$ was around 20 ps. When the RO started to be perturbed, $\sigma_{T_{osc}}$ increased, reached its maximum value of almost 50 ps and then dropped back to about 5 ps when the RO was locked. The values of $\sigma_{T_{osc}}$ and $\sigma_{T_{pert}}$ stayed low from 288 MHz to 296.5 MHz, thereby confirming that the lock-in range was the same as that obtained using the first method. The $\sigma_{T_{osc}}$ of natural ring oscillations is depicted as a reference.

3) $\overline{\Delta_\varphi}$: The third part of Fig. 7 shows the difference $\overline{\Delta_\varphi}$ between phases φ_{osc} and φ_{pert} measured thousand times using the oscilloscope. As detailed in Section IV-C3, when the oscillator was not locked to the perturbation signal, $\overline{\Delta_\varphi}$ was distributed uniformly between $-180\,°$ and $180\,°$. Therefore, $\overline{\Delta_\varphi}$ measured by the oscilloscope was centered around $0\,°$. On the contrary, when the ring was locked, $\overline{\Delta_\varphi}$ remained constant and different from zero. The evolution of the $\overline{\Delta_\varphi}$ value depending on f_{pert} confirmed the lock-in range giving a constant $\overline{\Delta_\varphi}$ equal to about $70\,°$ between 288 MHz to 293.5 MHz. However, between 293.5 MHz and 296.5 MHz, even if $\overline{\Delta_\varphi}$ had a constant value at each measurement, some phase inversions appeared: the $\overline{\Delta_\varphi}$ of $70\,°$ became $-110\,°$. The horizontal line represents the $\overline{\Delta_\varphi}$ of two independent signals.

4) σ_{Δ_φ}: Finally, the last part of Fig. 7 depicts σ_{Δ_φ} depending on f_{pert} measured using the oscilloscope. Except for some spikes due to the phase inversions explained above, the lock-in range was also confirmed with a σ_{Δ_φ} of $0\,°$ between 288 MHz and 296.5 MHz, and $104\,°$ otherwise. Again, the horizontal line represents the σ_{Δ_φ} of two independent signals.

The four methods detected the lock-in range of 8.5 MHz for the studied $N = 3$ RO implemented in the Xilinx Spartan 6 FPGA.

Since the identified lock-in range was the same using all four methods, only $\overline{\Delta f}$ is depicted in the rest of the paper. However, even if not depicted, the three other parameters were always computed to confirm the lock-in range identified with $\overline{\Delta f}$.

B. Study of the RO locking depending on N and on the family to which the device belongs

We extended our observations to two other FPGA families – Intel Cyclone V and Microsemi SmartFusion 2, since they are made using a different technology and their internal structure differs from that of Xilinx Spartan 6. Table I lists natural f_{osc}, lock-in range and N depending on the FPGA families. The results show that the smaller N (or the higher f_{osc}), the larger the lock-in range. Surprisingly, the relative lock-in range (in %) in the SmartFusion 2 family remained constant, while it was significantly higher at high frequencies in the other

two families. This effect can be explained by regular routing structures in the SmartFusion 2 family.

FPGA Family	N	Natural f_{osc} (MHz)	Lock-in range (MHz)	%
Xilinx Spartan 6	3	296	8.5	2.9
	7	144	2.1	1.4
	11	90	1	1.1
	15	67	0.6	0.9
	23	44	0.3	0.7
	31	33	0.2	0.6
Intel Cyclone V	7	409	5.8	1.3
	11	222	1	0.4
	15	144	0.45	0.3
	19	115	0.35	0.3
	21	106	0.3	0.3
	31	71	0.2	0.3
Microsemi Smartfusion 2	5	404	4.4	1.1
	7	266	3.3	1.2
	11	178	2.4	1.3
	15	123	1.6	1.3
	23	84	1.2	1.4
	31	63	0.9	1.4

TABLE I: Impact of locking on an RO depending on its natural f_{osc} and on the family to which the device belongs

To study the impact of f_{osc} on locking, a $N = 5$ RO was implemented in Xilinx Spartan 6 but placed in five different places inside the device to modify f_{osc}. The fastest RO was implemented with all inverters placed in the same configurable logic block (CLB) and the slowest one with one inverter per CLB. Figure 8 depicts the lock-in range depending on f_{osc}.

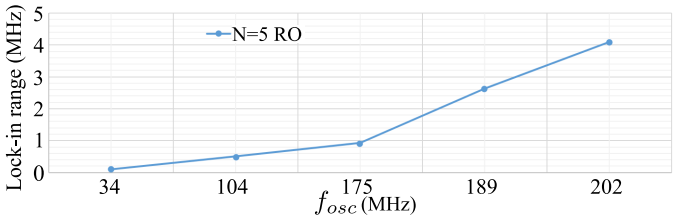


Fig. 8: Lock-in range depending on f_{osc} in the Xilinx Spartan 6 FPGA

From Table I and Fig. 8, it is clear that f_{osc} is the main parameter affecting the lock-in range. Since we are dealing with logic devices ($V_{pert} = V_{osc}$), Adler's equation that establishes the conditions for locking can be rearranged to:

$$\frac{f_{osc}}{2Q} > |f_{pert} - f_{osc}|, \tag{6}$$

thus confirming the results presented in Table I and Fig. 8; f_{osc} is the main parameter affecting the lock-in range.

C. RO locking on harmonic frequencies of the injected signal

In the following experiments, a $N = 3$ RO implemented in a Xilinx Spartan 6 FPGA was locked on the harmonic frequency of the injected signal. Its natural f_{osc} was 296 MHz. f_{pert} varied from 145 MHz to 150 MHz in steps of 0.025 MHz. The second harmonic of this signal was close to the natural f_{osc}. Figure 9 depicts the dependence of f_{osc} on f_{pert}. Frequency $f_{pert} \times 2$ represents the second harmonic of the perturbation signal. Results show that with a lock-in range from 146.5 MHz to 148.5 MHz, RO locks to harmonic frequencies of the perturbation signal. However, the lock-in range is narrower because of the lower frequency of the perturbation signal f_{pert}.

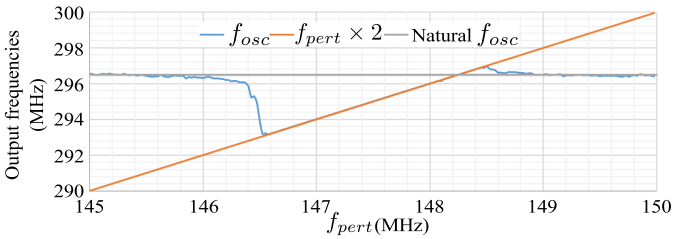


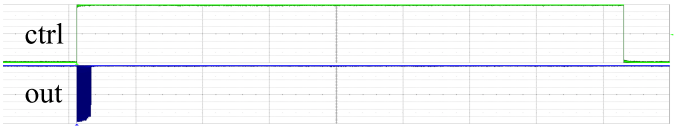
Fig. 9: Evolution of f_{osc} of a $N = 3$ RO depending on f_{pert} in Xilinx Spartan 6 FPGA

VI. Locking effects on transient effect ring oscillators

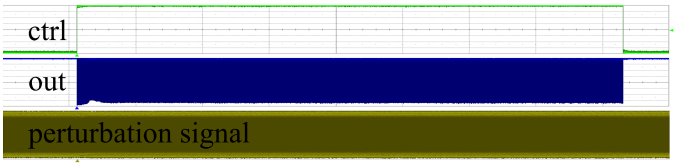
A. TERO frequency perturbation using a delay line

Next, the same kind of experiments as those presented in the previous section were performed, but on a $N = 5$ TERO implemented in Xilinx Spartan 6 FPGA.

The TERO natural f_{osc} was 196 MHz with a mean number of oscillations after excitation $N_{osc} = 85$. Figure 10a shows the TERO output signal (*out*) and the activation signal (*ctrl*) when no perturbation was sent into the delay line. Figure 10b shows the same TERO with perturbations. While when not perturbed, the TERO stopped as expected, with perturbations, the TERO behaved differently – it tended to oscillate permanently. Indeed, as can be seen in Fig. 10b, oscillations persisted until the control signal went to zero.



(a) TERO output with no perturbation



(b) TERO output in the presence of a perturbation signal

Fig. 10: TERO output without and with perturbation

Figure 11 depicts the evolution of the duty cycle of the TERO output signal for 100 acquisitions with no perturbation. As explained above, in nominal conditions, the duty cycle gradually changes from around 50% towards 0% or 100% until oscillations stop at the low or high logical level, respectively.

Both cases can be represented by the parameter D expressed in %:

$$D = 50 - |50 - t_h| = 50 - |50 - t_l|\,, \tag{7}$$

where t_h and t_l represent the percentage of the clock period, during which the TERO output is at a logical high or a logical low level, respectively.

Figure 12 shows the evolution of parameter D for the same TERO with perturbations. The maximum number of oscillations presented in the figure ($N_{osc} \sim 450$) is determined by the acquisition window of the oscilloscope, but it is clear that D always stays close to 50% and oscillations persist even after acquisition. It is also important to note that even when D starts to change, the two signals can sometimes lock again, pulling D back to its initial value.

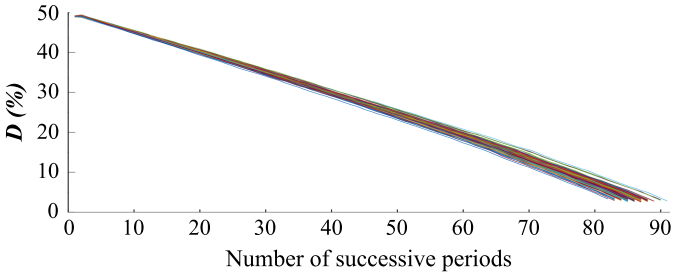


Fig. 11: Evolution of the duty cycle expressed using parameter D for the unperturbed TERO

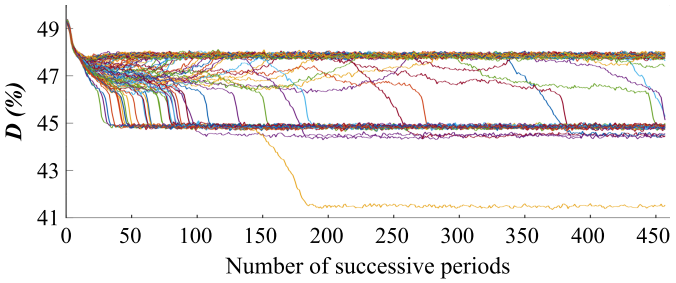


Fig. 12: Evolution of the duty cycle expressed using parameter D for the perturbed TERO

In addition to using the N_{osc} as evidence for the locking phenomenon in TERO, the four other lock-in detection methods introduced in Section IV-C were also used. They confirmed that TEROs can lock to perturbation signals just like ROs do. The measured lock-in range for the $N = 5$ TERO was $2.8\,\mathrm{MHz}$.

B. TERO self-locking

As explained in Section III-B, two electrical events propagate across the TERO cells while it is oscillating. Due to variations in the CMOS manufacturing process, one event (e.g. the rising or falling edge) is faster than the other one (falling or rising edge) and the oscillations last until both events collapse. Thus, parameter D shifts from around 50% towards 0% as shown in Fig. 11.

However, in some cases (*e.g.* when the two branches of the TERO are too close to each other or even interleaved), the two events can lock to each other and the parameter D undergoes no further changes.

Figure 13 depicts changes in parameter D for a $N = 3$ TERO implemented in the Xilinx Spartan 6 FPGA, when the TERO tends to lock to itself. At some moment, D stops changing and remains constant at a value of about 25%. If the ring does not lock to itself, the mean number of oscillations is $N_{osc} \sim 600$, but when the two events lock to each other, N_{osc} can become very high.

Moreover, it may happen that the two events remain locked for a certain time and then suddenly unlock, triggering the end of oscillations (see Fig. 13 near N_{osc} equal to 4000 or 11000, for which D drops to 0).

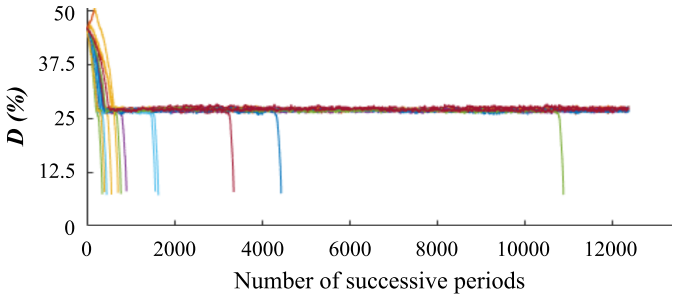


Fig. 13: Evolution of the duty cycle using parameter D for the TERO locked to itself

C. Study of the TERO locking depending on N and on the family to which the device belongs

Like in experiments with ROs, we studied the effect of locking in TEROs depending on N in the three FPGA families. The results in Table II show that like ROs, the f_{osc}, which is determined by N, has a significant impact on the locking phenomenon.

FPGA Family	***N***	**Natural f_{osc}** (MHz)	**Lock-in range** (MHz)	**%**
Xilinx Spartan 6	3	317	6.4	2
	15	69	0.5	0.7
	31	33	0.1	0.3
Intel Cyclone V	3	466	8	1.7
	5	258	3.4	1.3
	7	188	1.2	0.6
Microsemi Smartfusion 2	5	293	3.2	1.1
	15	117	1.5	1.3
	31	65	0.4	0.6

TABLE II: Impact of locking on a TERO depending on its natural f_{osc} and on the family to which the device belongs

However, what is specific to TERO, is that the N_{osc} also has some impact on the lock-in range. Figure 14 shows the lock-in range of three $N = 3$ TEROs implemented in the Xilinx Spartan 6 FPGA. The three TEROs had the natural $f_{osc} = 350\,\mathrm{MHz}$ and different N_{osc}. The figure shows that when N_{osc} increases, the lock-in range also increases.

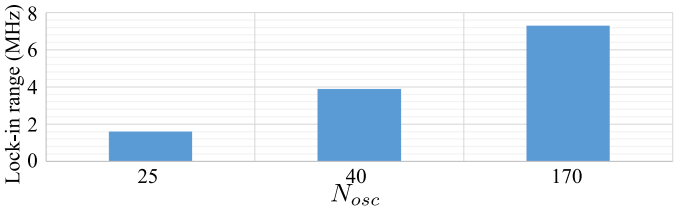


Fig. 14: Lock-in range for different N_{osc} of $N = 3$ TEROs implemented in the Xilinx Spartan 6 FPGA

D. TERO locking on harmonic frequencies of the injected signal

In the following experiment, the second harmonic of a $N = 5$ TERO implemented in Xilinx Spartan 6 was locked on the frequency of the injected signal. The TERO natural f_{osc} was 196 MHz with a mean N_{osc} of 85 after excitation. f_{pert} varied from 384 MHz to 398 MHz with a step of 0.025 MHz. Figure 15 depicts the dependence of f_{osc} over f_{pert}. $f_{pert}/2$ represents the 1/2 output signal frequency of the perturbation signal. Results show that the second harmonic of the f_{osc} locks to the f_{pert} with a lock-in range from 387 MHz to 391 MHz. The locking frequency is even more efficient with harmonic perturbation due to a twice bigger f_{pert}.

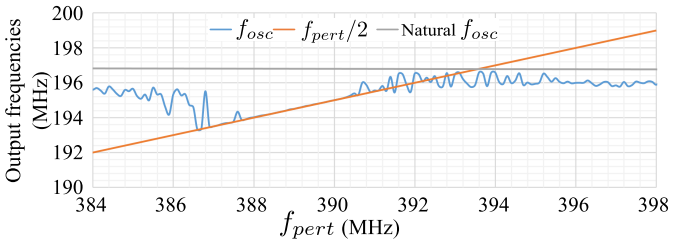


Fig. 15: Evolution of f_{osc} of a $N = 5$ TERO depending on f_{pert} in Xilinx Spartan 6

VII. Locking effects on self-timed ring oscillators

A. STR frequency perturbation using a delay line

As explained in Section III-C, STR can operate in one of two modes, an evenly-spaced mode and a burst mode. When operating in the evenly-spaced mode, the mean f_{osc} is stable. When STRs operate in the burst mode, several short periods that arrive in a burst are followed by a time interval during which no events occur. For more details about STR operation modes, the reader can refer to [26]. In our experiments, the STR cells were loaded with $N/2$ events, guaranteeing that the ring always operated in the evenly-spaced mode.

The locking phenomenon was evaluated on a $N = 8$ STR implemented in the Xilinx Spartan 6 FPGA. Without being perturbed, the natural f_{osc} of the ring was 336 MHz. Next, the f_{pert} varied from 323 MHz to 344 MHz with steps of 0.025 MHz. As can be seen in Fig. 16, the STR locked to the perturbation signal just like other rings with a lock-in range of about 12.5 MHz.

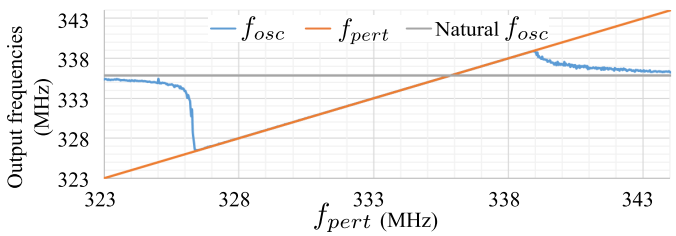


Fig. 16: Evolution of f_{osc} of a $N = 8$ STR depending on f_{pert} in Xilinx Spartan 6

B. Study of the STR locking depending on N and on the family to which the device belongs

Like for the previous two types of rings, we studied the effect of locking in STRs depending on N in the three FPGA families. The results are presented in Table III. Again, the f_{osc} has a significant impact on the locking phenomenon.

FPGA Family	***N***	**Natural f_{osc}** (MHz)	**Lock-in range** (MHz)	**%**
Xilinx Spartan 6	8	334	12.5	3.7
	16	334	12.4	3.7
	32	299	10.5	3.5
	64	263	4.8	1.8
Intel Cyclone V	8	420	2.3	0.5
	16	396	2.1	0.5
	32	383	0.7	0.2
	64	368	0.5	0.1
Microsemi Smartfusion 2	8	442	1.4	0.3
	16	392	1.4	0.3
	32	369	1	0.3
	64	255	1.1	0.4

TABLE III: Impact of locking on an STR depending on its natural f_{osc} and on the family to which the device belongs

C. STR locking on harmonic frequencies of the injected signal

In our last experiment, like those with ROs and TEROs, a $N = 8$ STR implemented in the Xilinx Spartan 6 FPGA was locked on the harmonic frequency of the injected signal.

The natural f_{osc} of the STR was 334 MHz. f_{pert} varied from 164 MHz to 169 MHz with steps of 0.025 MHz. Figure 17 shows dependence of f_{osc} on f_{pert}. Frequency $f_{pert} \times 2$ represents the second harmonic of the perturbation signal. The lock-in range obtained was 2.1 MHz.

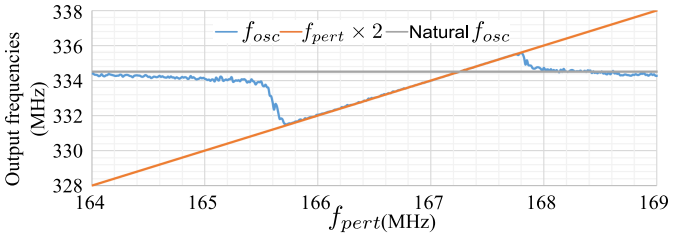


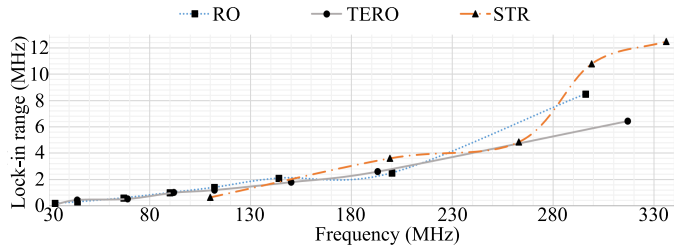
Fig. 17: Evolution of f_{osc} of a $N = 8$ STR depending on f_{pert} in Xilinx Spartan 6

VIII. Summary and comparison of results

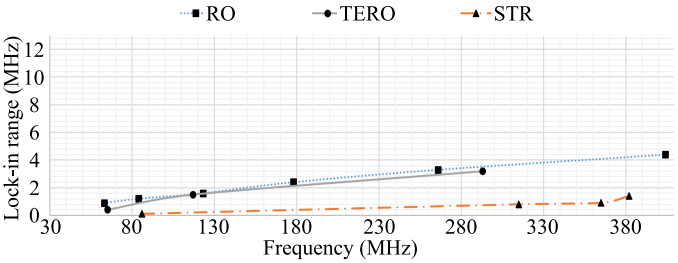
A. Comparison of the types of rings studied

Figure 18 shows the lock-in range of ROs, TEROs and STRs depending on their f_{osc} on Xilinx Spartan 6 and Microsemi SmartFusion 2. The results for all kinds of rings are very similar. We can thus conclude that independently of the type of ring, all the rings we studied could lock to the perturbation signal or its harmonics and the lock-in range increased with the frequency.

It is also worth noting that STRs tend to be the most sensitive to locking with an increase in frequency when implemented in the Spartan 6 FPGA (see Fig. 18a). However, when implemented in the SmartFusion2 FPGA, the opposite is the case (see Fig. 18b). This demonstrates that the type of FPGA and its technology do have an influence on the locking impact thereby making it difficult to compare rings on FPGA.



(a) Lock-in range of the RO, TERO and STR in the Xilinx Spartan 6 FPGA



(b) Lock-in range of the RO, TERO and STR in the Microsemi SmartFusion2 FPGA

Fig. 18: Comparison of lock-in ranges of the RO, TERO and STR implemented in Spartan 6 and SmartFusion2 FPGA families

B. Comparison of the locking phenomenon in different FPGA families

Using the same approach and comparing the impact of the locking phenomenon on ROs implemented in the three FPGA families presented in Fig. 19 shows that, whatever the device used, locking has an impact on the rings. Based on these results, the Xilinx Spartan 6 FPGA appears to be the most sensitive to locking. It is probable that this sensitivity comes from the routing resources available and the way the rings were routed. However, it is not guaranteed that repeating the same experiment with rings placed in another area of the FPGA would have similar results. We analyze the effect of routing in the following section.

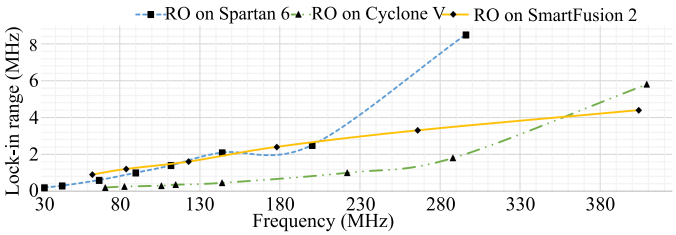


Fig. 19: Comparison of lock-in ranges of ROs implemented in Xilinx Spartan 6, Intel Cyclone V and Microsemi SmartFusion2 FPGA families

IX. Effect of placement and routing on the locking phenomenon

In experiments to study the effect of placement and routing (P/R) of rings, a $N = 7$ RO was implemented in the Intel Cyclone V FPGA. The ring was placed in the same place in all the experiments, but the perturbation delay line was placed in five different ways. The ring always oscillated at a natural f_{osc} of about $410\,\mathrm{MHz}$. Figure 20 shows screen-shots of the Intel chip planner tool for the five P/R configurations. The RO elements are shown in red and the delay line elements in yellow.

In the first P/R configuration (P/R1 presented in Fig. 20a), elements of the RO and elements of the delay line were placed, as for all the previous experiments (see Fig. 6), interleaved in the same adaptive logic module (ALM).

In the second P/R configuration (P/R2 presented in Fig. 20b), elements of the RO and elements of the delay line were not in the same ALM, but still in the same logic array block (LAB).

In the third P/R configuration (P/R3 depicted in Fig. 20c), the delay line was composed of only one inverter.

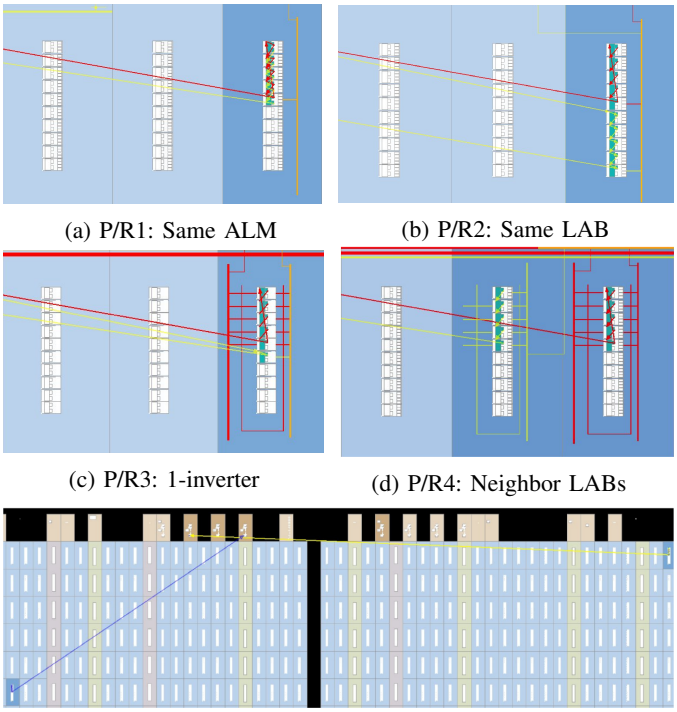
In the fourth P/R configuration (P/R4 in Fig. 20d), the delay line was placed in the neighboring LAB.

Finally, in the fifth P/R configuration (P/R5 in Fig. 20e), the delay line was placed in the opposite corner of the FPGA.

Figure 21 shows the lock-in range of the $N = 7$ RO for the five different delay line placements in Intel Cyclone V. It is clear that the closer the RO elements to the delay line elements, the larger the lock-in range.

Indeed, in P/R1, the elements are placed in the same ALM and the resulting lock-in range is $4.4\,\mathrm{MHz}$. In P/R2 and P/R3, the elements are still in the same LAB but not in the same ALM and the lock-in is $3\,\mathrm{MHz}$ smaller. It gets even smaller with P/R4, when the elements are in neighboring LABs, to end up with no locking in the P/R5, when the elements are placed in the opposite corner of the FPGA.

The results show a clear correlation between the proximity of the rings and their capacity to lock. When they are closer to one another, the rings use more routes and the EM interferences are stronger.



(a) P/R1: Same ALM (b) P/R2: Same LAB

(c) P/R3: 1-inverter (d) P/R4: Neighbor LABs

(e) P/R5: Opposite corner of the FPGA

Fig. 20: Configurations of P/R of the $N = 7$ RO and the delay line in Intel Cyclone V FPGA

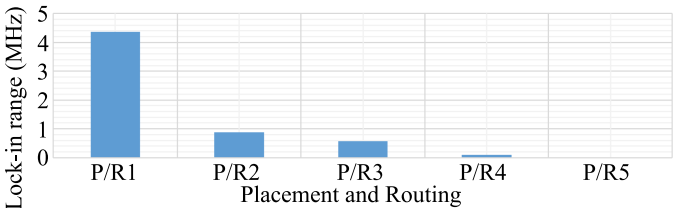


Fig. 21: Lock-in ranges of a $N = 7$ RO depending on the P/R of the delay line in the Intel Cyclone V FPGA

X. Use cases of oscillating rings and impact of locking on their operation

In this section, we extend our study to two use cases using oscillating rings:

- A TERO based TRNG operating in an environment with the presence of a perturbation signal.
- A system using two ROs oscillating at close frequencies in an environment with voltage variations.

A. Impact of locking on the TERO-TRNG

TRNGs are used in cryptography to generate confidential keys, initialization vectors, challenges, nonces in cryptographic protocols, and random masks in side channel attack countermeasures. They exploit intrinsic noise sources in electronic devices as a source of randomness. TRNGs aimed at cryptographic applications must fulfill security requirements defined by standards AIS-20/31 [32] or FIPS [33].

The TERO based TRNG depicted in Fig. 22 was first published in [34]. The TERO, which generates random numbers of oscillation periods is followed by a counter and an output data register. The counter counts the number of oscillation periods and its least significant bit represents the TRNG output. For the sake of simplicity, higher bits of the counter are omitted in Fig. 22. The control signal, which periodically restarts the TERO, determines the output bit rate of the generator.

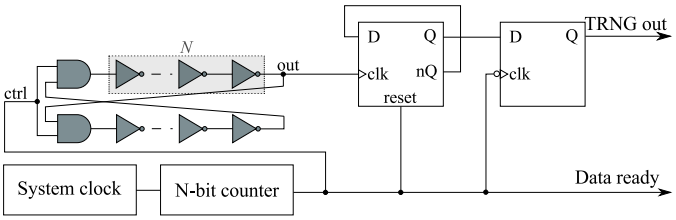
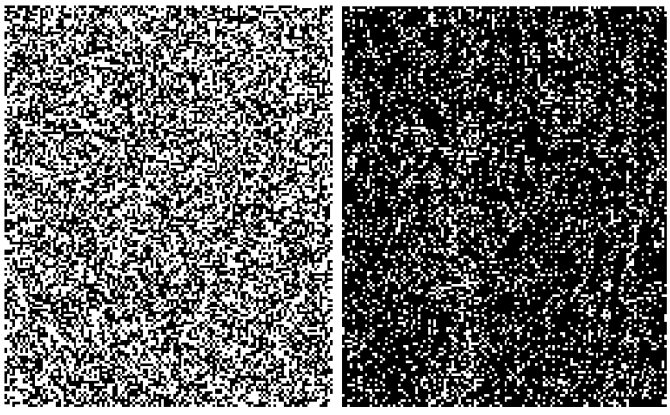


Fig. 22: Block diagram of the TERO-based TRNG

Like in the previous experiments, the elements of the delay line, to which the perturbation signal was sent, was interleaved with the TERO inverters. Two bit-streams of 1 million bits were acquired: one without any perturbation signal and one with the TERO locked to the perturbation signal. The generated numbers were tested using the FIPS 140-1 standard test suite [33]. While without perturbation, data passed the statistical tests, data generated with the perturbed TRNG failed to pass the tests.

The impact of locking on the TERO-TRNG can be seen in Fig. 23a and 23b, in which black pixels represent logical zero and white pixel represent logical one. Without perturbation, the number of zeros and ones is approximately the same, i.e. the generator output is unbiased. On the contrary, when the TERO cell is locked to the perturbation signal (i.e. it tends to oscillate permanently after each excitation), the TRNG output is biased towards zeros, which causes generation of the random bit stream featuring a low entropy rate.



(a) Without perturbation (b) Under perturbation

Fig. 23: TERO-based TRNG output data without and with perturbation

B. Effect of mutual locking of two clock signals generated in similar ROs

In the second use case, instead of perturbing a RO by some perturbation signal, we implemented two similar ROs, which

could be used, for example, as clock generators. The two ROs were oscillating at close frequencies.

Indeed, in nominal operating conditions the two ROs oscillated freely and no locking effect occurred. However, as stated in Section I, oscillating rings are very sensitive to voltage variations and their natural f_{osc} increases with the power voltage, but not exactly in the same way for all rings. In our experiment, we modified the power supply voltage in the permitted range from $960\,\mathrm{mV}$ to $1260\,\mathrm{mV}$.

Figure 24 depicts evolution of the f_{osc} of both ROs (f_{osc1} and f_{osc2}, respectively) depending on the voltage. It can be seen that the two rings locked to each other at voltages ranging from $1100\,\mathrm{mV}$ to $1140\,\mathrm{mV}$. Figure 24 shows also σ_{Δ_φ} of the first RO ($\sigma_{\Delta_{\varphi osc1}}$), which very clearly shows when the two ROs locked.

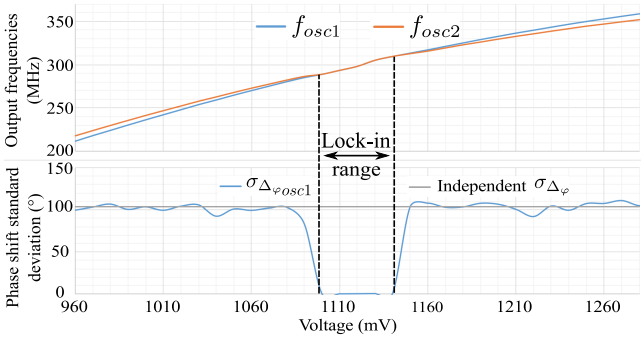


Fig. 24: Evolution of f_{osc1}, f_{osc2} and σ_{Δ_φ} depending on the power supply voltage of two ROs implemented in the Intel Cyclone V FPGA

XI. Conclusion and Further Work

In this paper, we presented, analyzed and discussed the locking phenomenon that occurs in oscillating rings. We showed for the first time that contrary to other claims, the structures of new kinds of oscillating rings, like TERO and STR, are also sensitive to locking phenomena.

By showing the presence of the locking effect on three FPGA families from three different manufacturers, we demonstrated that the locking phenomenon occurs independently of the family to which the device belongs.

The results experimentally confirm theoretical papers of Adler, Tucker and Huntoon. Indeed, according to Adler's equation and because the study is performed on digital circuits ($V_{pert} = V_{osc}$): the higher the f_{osc}, the larger the lock-in range. This behavior was confirmed in all our experiments.

It is important to stress that the proximity of the oscillating rings and their routing significantly impact the effect of the locking phenomenon. Unfortunately, the routing is very often very difficult to control in FPGAs. Therefore, it would be better to study the impact of routing on the locking phenomenon in some dedicated integrated circuit in which the routing could be fully controlled.

Finally, we exposed the disastrous consequences of locking on two use cases based on oscillating rings: a TERO TRNG and a system using multiple clocks generated using oscillating rings.

The results presented here, together with the freely available open source VHDL codes, will help designers better anticipate locking phenomenon in their future designs. We believe that most of our conclusions can be extended to implementations of oscillating rings in application specific integrated circuits (ASICs).

Acknowledgments

This work was carried out in the framework of the FUIAAP22-Project PILAS supported by Bpifrance.

The authors are also grateful to Kamil Caglar for his experiments on the different rings.

References

[1] J. F. Ewen, A. X. Widmer, M. Soyuer, K. R. Wrenner, B. Parker, and H. A. Ainspan, "Single-chip 1062 Mbaud CMOS transceiver for serial data communication," in *Solid-State Circuits Conference, 1995. Digest of Technical Papers. 41st ISSCC, 1995 IEEE International*, Feb 1995, pp. 32–33.

[2] T. A. D. Riley, M. A. Copeland, and T. A. Kwasniewski, "Delta-sigma modulation in fractional-N frequency synthesis," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 5, pp. 553–559, May 1993.

[3] A. Oustaloup, Y. Deval, D. Belot, P. Melchior, J. Begueret, F. Badets, and V. Lagareste, "PLL-based frequency synthesizer," Jun. 29 2006, uS Patent App. 11/235,787.

[4] M. Baudet, D. Lubicz, J. Micolod, and A. Tassiaux, "On the security of oscillator-based random number generators," *Journal of Cryptology*, vol. 24, no. 2, pp. 398–425, 2011.

[5] B. Sunar, W. J. Martin, and D. R. Stinson, "A provably secure true random number generator with built-in tolerance to active attacks," *IEEE Transactions on Computer*, vol. 56, no. 1, pp. 109–119, Jan. 2007.

[6] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Silicon physical random functions," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, ser. CCS '02. New York, NY, USA: ACM, 2002, pp. 148–160.

[7] L. Bossuet, X. T. Ngo, Z. Cherif, and V. Fischer, "A PUF based on a transient effect ring oscillator and insensitive to locking phenomenon," *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 1, pp. 30–36, 2014.

[8] D. Chattopadhyay and P. Rakshit, *Electronics (Fundamentals And Applications)*. New Age International, 2006, ch. 11, pp. 224 – 246.

[9] O. El Issati, "Self-Timed Ring Oscillators : from theory to practice," PhD Thesis, Institut National Polytechnique de Grenoble - INPG, Sep. 2011.

[10] M. Mandal and B. C. Sarkar, "Ring oscillators: Characteristics and applications," *Indian Journal of Pure and Applied Physics*, vol. 48, pp. 136–145, 02 2010.

[11] T. C. Weigandt, B. Kim, and P. R. Gray, "Analysis of timing jitter in CMOS ring oscillators," in *Proceedings of IEEE International Symposium on Circuits and Systems - ISCAS '94*, vol. 4, May 1994, pp. 27–30 vol.4.

[12] A. A. Abidi, "Phase noise and jitter in CMOS ring oscillators," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 8, pp. 1803–1816, Aug 2006.

[13] C.-C. Chen, W.-J. Liu, S.-H. Lin, and C.-C. Lin, "A CMOS oscillators-based smart temperature sensor for low-power low-cost systems," in *Procedia Engineering*, vol. 47, 12 2012, p. 9295.

[14] B. Mesgarzadeh and A. Alvandpour, "A study of injection locking in ring oscillators," in *2005 IEEE International Symposium on Circuits and Systems*, May 2005, pp. 5465–5468 Vol. 6.

[15] R. J. Betancourt-Zamora, S. Verma, and T. H. Lee, "1-GHz and 2.8-GHz CMOS injection-locked ring oscillator prescalers," in *2001 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No.01CH37185)*, June 2001, pp. 47–50.

[16] 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*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 151–166.

[17] A. Mirzaei, M. E. Heidari, R. Bagheri, and A. A. Abidi, "Multi-phase injection widens lock range of ring-oscillator-based frequency dividers," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 3, pp. 656–671, March 2008.

[18] J. C. Chien and L. H. Lu, "Analysis and design of wideband injection-locked ring oscillators with multiple-input injection," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 9, pp. 1906–1915, Sept 2007.

[19] N. Bochard, F. Bernard, V. Fischer, and B. Valtchanov, "True-Randomness and Pseudo-Randomness in Ring Oscillator-Based True Random Number Generators," *International Journal of Reconfigurable Computing*, vol. 2010, p. ID 879281, Dec. 2010, 12 pages.

[20] F. Bernard, V. Fischer, C. Costea, and R. Fouquet, "Implementation of ring-oscillators-based physical unclonable functions with independent bits in the response," *International Journal of Reconfigurable Computing*, vol. 2012, pp. 13:13–13:13, Jan. 2012.

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

[22] L. M. Reyneri, D. D. Corso, and B. Sacco, "Oscillatory metastability in homogeneous and inhomogeneous flip-flops," *IEEE Journal of Solid-State Circuits*, vol. 25, no. 1, pp. 254–264, Feb 1990.

[23] A. Cherkaoui, L. Bossuet, and C. Marchand, "Design, evaluation, and optimization of physical unclonable functions based on transient effect ring oscillators," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1291–1305, June 2016.

[24] C. Marchand, L. Bossuet, U. Mureddu, N. Bochard, A. Cherkaoui, and V. Fischer, "Implementation and characterization of a physical unclonable function for IoT: A case study with the TERO-PUF," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 1, pp. 97–109, Jan 2018.

[25] D. Muller and W. Bartky, "A theory of asynchronous circuits," *Proc. Int'l Symp. Theory of Switching, Part 1, Harvard Univ. Press*, pp. 204–243, 1959.

[26] J. Hamon and L. Fesquet, "Robust and programmable Self-Timed Ring oscillators," in *IEEE 9th International New Circuits and Systems Conference (NEWCAS)*, 2011, pp. 249–252.

[27] H. Oliveira and L. Viseu Melo, "Huygens synchronization of two clocks," *Scientific Reports*, vol. 5, no. 11548, 06 2015.

[28] D. G. Tucker, "Forced oscillations in oscillator circuits, and the synchronization of oscillators," *Electrical Engineers - Part III: Radio and Communication Engineering, Journal of the Institution of*, vol. 92, no. 19, pp. 226–234, September 1945.

[29] R. Adler, "A study of locking phenomena in oscillators," *Proceedings of the IRE*, vol. 34, no. 6, pp. 351–357, June 1946.

[30] R. D. Huntoon and A. Weiss, "Synchronization of oscillators," *Journal of Research of the National Bureau of Standards*, vol. Volume 38, pp. 397–410, April 1947.

[31] N. Bochard, C. Marchand, O. Petura, L. Bossuet, and V. Fischer, "Evariste III: A new multi-FPGA system for fair benchmarking of hardware dependent cryptographic primitives," *Workshop on Cryptographic Hardware and Embedded Systems CHES*, Sep 2015, poster.

[32] W. Killmann and W. Schindler, *A proposal for: Functionality classes for random number generators, version 2.0*, BSI Std., 2011. [Online]. Available: https://www.bsi.bund.de/EN/Home/home_node.htm

[33] *FIPS 140-1: Security Requirements for Cryptographic Modules*, NIST Std., 1994. [Online]. Available: http://csrc.nist.gov/publications/fips/fips140-1/fips1401.pdf

[34] M. Varchola and M. Drutarovsky, "New high entropy element for FPGA based true random number generators," *Workshop on Cryptographic Hardware and Embedded Systems CHES*, pp. 351–365, 2010.

Ugo Mureddu received the M.Sc. degree (2015) in electronics and embedded systems from "Institut National des Sciences Appliquées", Lyon, France and the M.Sc. degree (2015) in embedded systems and telecommunication engineering from "Telecom Saint-Etienne", Saint-Etienne, France. He is currently a third year Ph.D. student in Hubert Curien Laboratory, University of Lyon. His main research activities focus on secure embedded systems, IC security, and especially True Random Number Generators and Physical Unclonable Functions embedded in logic devices.

Ing. Nathalie Bochard received the master's degree in electronic engineering in 1996 and the Diploma of Technological Research (DRT) in vision, telecommunication and instrumentation from the University of Lyon, in 1997. She is a research engineer at the CNRS (National Center for Scientific Research) in France, which she joined in 1998. Currently, her main research interests include embedded hardware cryptographic architectures for configurable logic devices and especially design, implementation and evaluation of true random number generators and physical unclonable functions aimed at cryptographic applications. She is a senior member of the CryptArchi club.

Prof. Lilian Bossuet received the M.Sc. degree in electrical engineering from INSA, Rennes, France in 2001, and his Ph.D. in electrical engineering and computer sciences from the University of South Britanny, Lorient, France in 2004. From 2005 to 2010, he was Associate Professor, and head of the Embedded System Department at Bordeaux Institute of Technology. From 2010 to 2017, he was Associate Professor at the University of Lyon/Saint-Etienne and currently holds the special CNRS (Centre National de la Recherche Scientifique) Chair of Applied Cryptography and Embedded System Security. Since 2017, he has been Professor at the University of Lyon/Saint-Etienne, here he is head of the computer science department of the Hubert Curien Laboratory, he is also head of the secured embedded systems and hardware architecture group of this laboratory. In 2016, he received the General Ferri Award in electronics from the French SEE for his contribution to protection against IC counterfeiting and the IP protection. His main research focus is the security of embedded systems, IP protection, PUF design and characterization, secure-by-design crypto-processor, and reconfigurable architecture. L. Bossuet has published over 150 refereed publications in these areas and is a senior member of the IEEE.

Prof. Viktor Fischer received his M.S. and Ph.D. degrees in Electrical Engineering from Technical University of Kosice in Slovakia. From 1981 to 1991 he held an Assistant Professor position at the Department of Electronics of the Technical University of Kosice. From 1991 to 2006 he was a part-time invited professor at the University of Saint-Etienne, France. From 1999 to 2006 he was also a consultant with Micronic Slovakia, oriented in hardware data security systems. From 2006 he is a full-time Professor at the University of Saint-Etienne. His research interests include cryptographic engineering, secure embedded systems, cryptographic processors and especially true random number generators embedded in logic devices. He is a senior member of the CryptArchi club.