On True Random Number Generators Based on Chaotic Oscillations in General and Their Implementation on FPGAs in Particular

Markus Dichtl

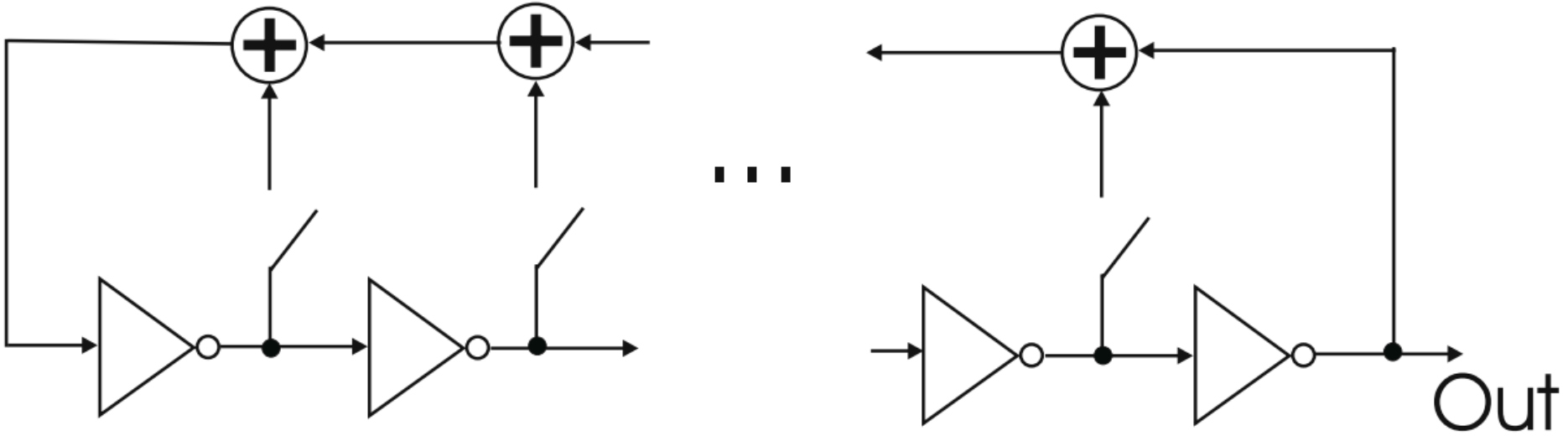
My first encounter with chaotic oscillations in 2006

- When reading Jovan Golić's paper New Methods for Digital Generation and Postprocessing of Random Data, IEEE Transactions on Computers (Volume: 55, Issue: 10, October 2006) on FIROs and GAROs for the first time I was very convinced that his claims were to good to be true, and that what he described was just pseudorandom behaviour in the style of LFSRs.

- So I came up with the restarting approach (at that time I assumed it was always from the same initial state).

- I went to the lab immediately and implemented the design on a Spartan 3 FPGA. The output was connected to an oscilloscope. I was really puzzled to see strongly varying behaviour just 25 ns or so after the restart. I concluded incorrectly that this proved truly random behaviour.

What is a FIRO?



What are Chaotic Oscillations?

- They are a mathematical concept defined for deterministic continuous systems
- The time of this talk could easily be spent discussing various aspects of the various definitions (but no, do not be afraid)
- Is it a good idea to base the design of a TRNG on the very complicated behaviour of a deterministic system?

Chaotic systems in very vague terms (Sorry!)

- Sensitive to initial conditions
- Topologically transitive
- Dense periodic orbits

Is it really chaotic?

- The initial question should be: Chaotic on which set of states?
- Despite my search for it, I did not find but very simple examples for which the required mathematical properties were shown.
- Should be really difficult to characterize which states a FIRO, say, can assume.
- Should be even more difficult to prove the required properties for chaos

A clear opinion

Probably the most objectionable principle for physical generation of random numbers is to obtain them from repeated measurements of a physical system in chaos. The philosophical problem here is that chaos assumes the existence of an underlying order in what is seemingly random. So why would someone knowingly make use of a nonrandom system in order to generate random numbers? We are not aware of anyone so far asking or answering this question.

- Mario Stipcević and Çetin Kaya Koç. “True Random Number Generators” in Open Problems in Mathematics and Computational Science, Editor Çetin Kaya Koç, Springer 2014

My view

- Of course, chaotic TRNGs exhibit noise
- Chaotic TRNGs are hoped to accumulate jitter contributions exponentially fast, in contrast with the very poor jitter accumulation of e. g. ROs

From another perspective

- In principle, it is a bad idea just to add small statistically independent jitter contributions numerically, as it takes n^2 summands to increase the standard deviation by a factor of n (central limit theorem).
- It seems be be a good idea to make copies of signals with jitter as soon as possible, as this reduces the risk of one jitter contribution to be cancelled by another one in the opposite direction.
- But of course, these signals will also be disappear from the system eventually...

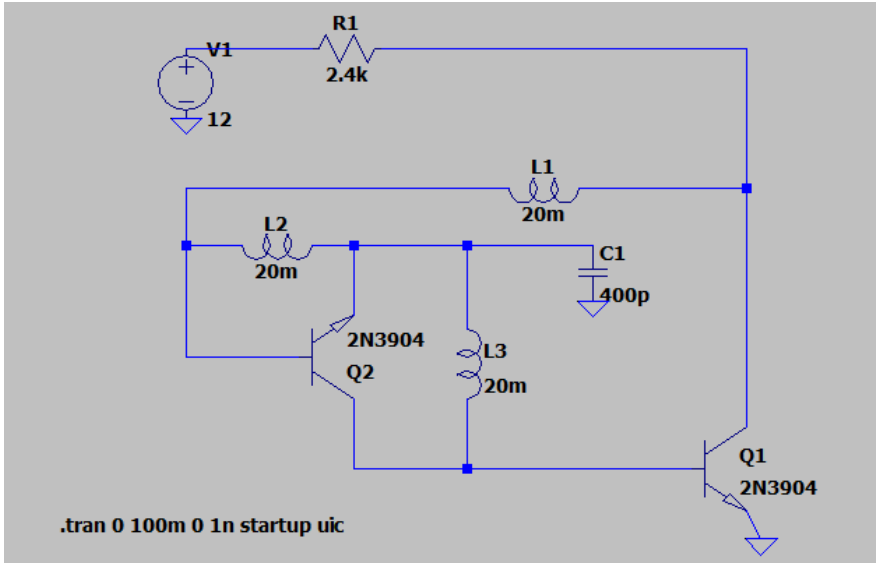
Two publications with very strange circuits

- Minati, Frasca, Oświęcimka, Faes, Drożdż: Atypical transistor-based chaotic oscillators: Design, realization, and diversity, Chaos 27 (2017)

  This approach to circuit design is rather extraordinary: They just simulate all possible ways to connect a small set of electronic components. If the design oscillates, they modify the parameters in order to find a configuration which oscillates chaotically.

- J. Scott, W. Thio: Elegant Circuits Simple Chaotic Oscillators, World Scientific 2021

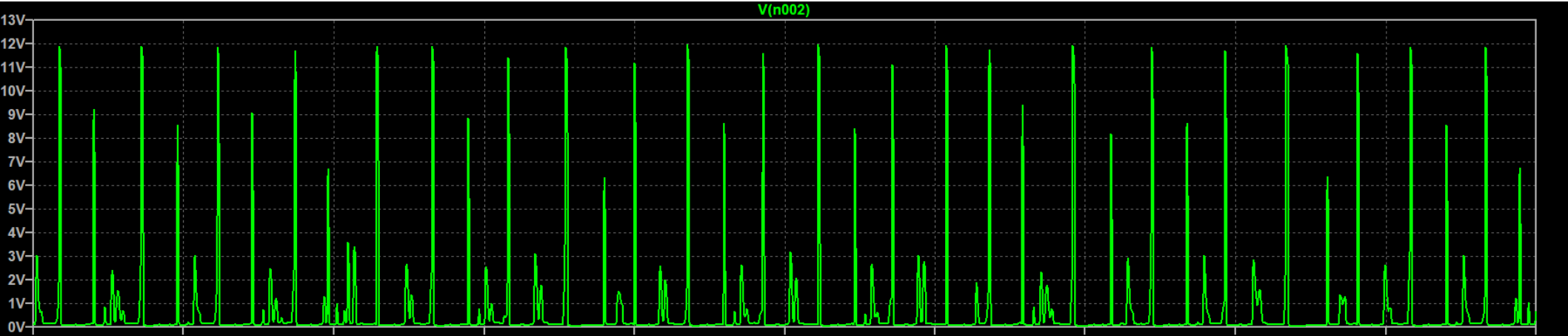
A simulated chaotic oscillation from a strange circuit



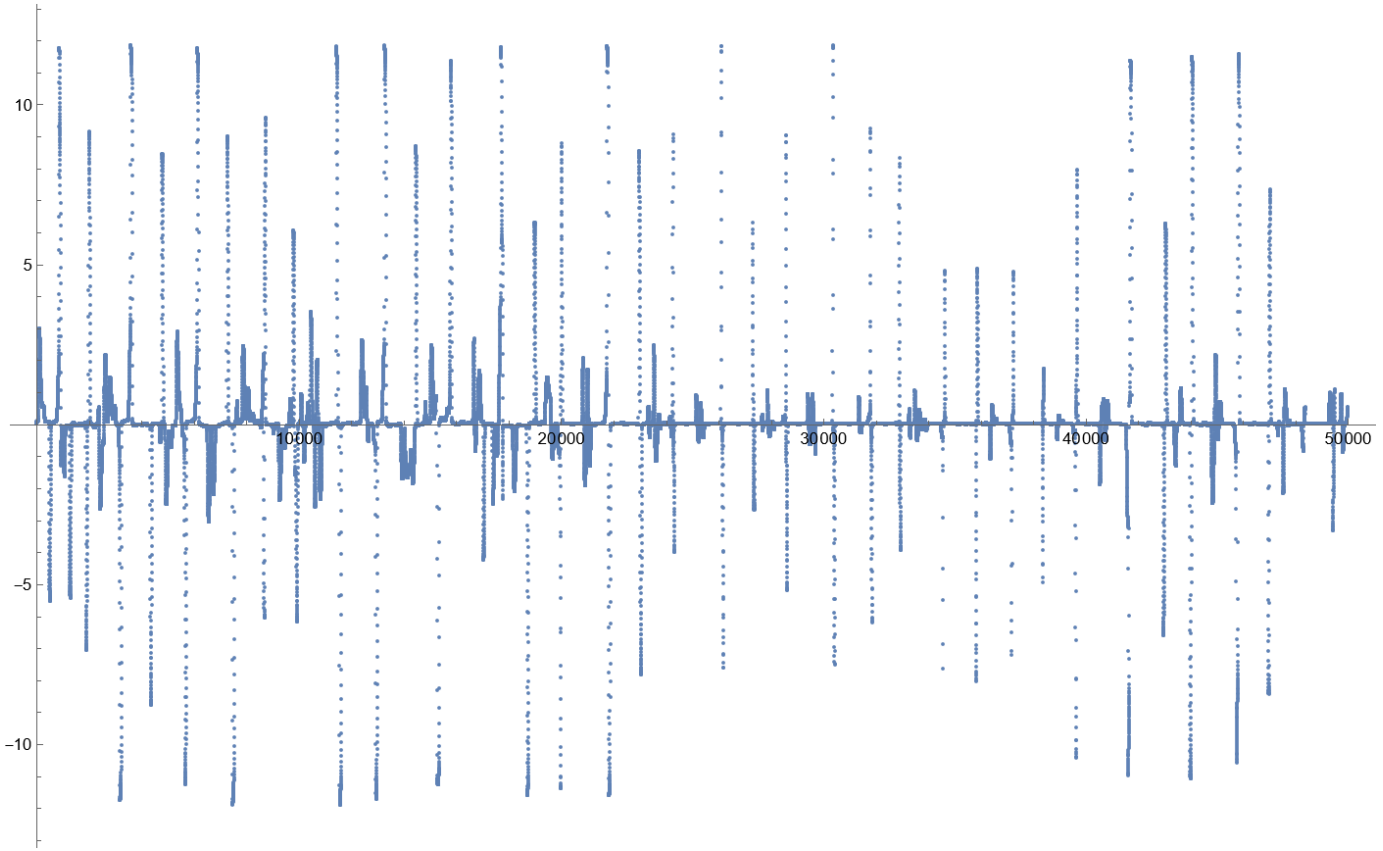
Scott-Thio variant of the spiking Minati et al. circuit. My choice of parameters.

Maximum correlation of this 0.5 ms piece with the 9900000 previous ones: 0.417

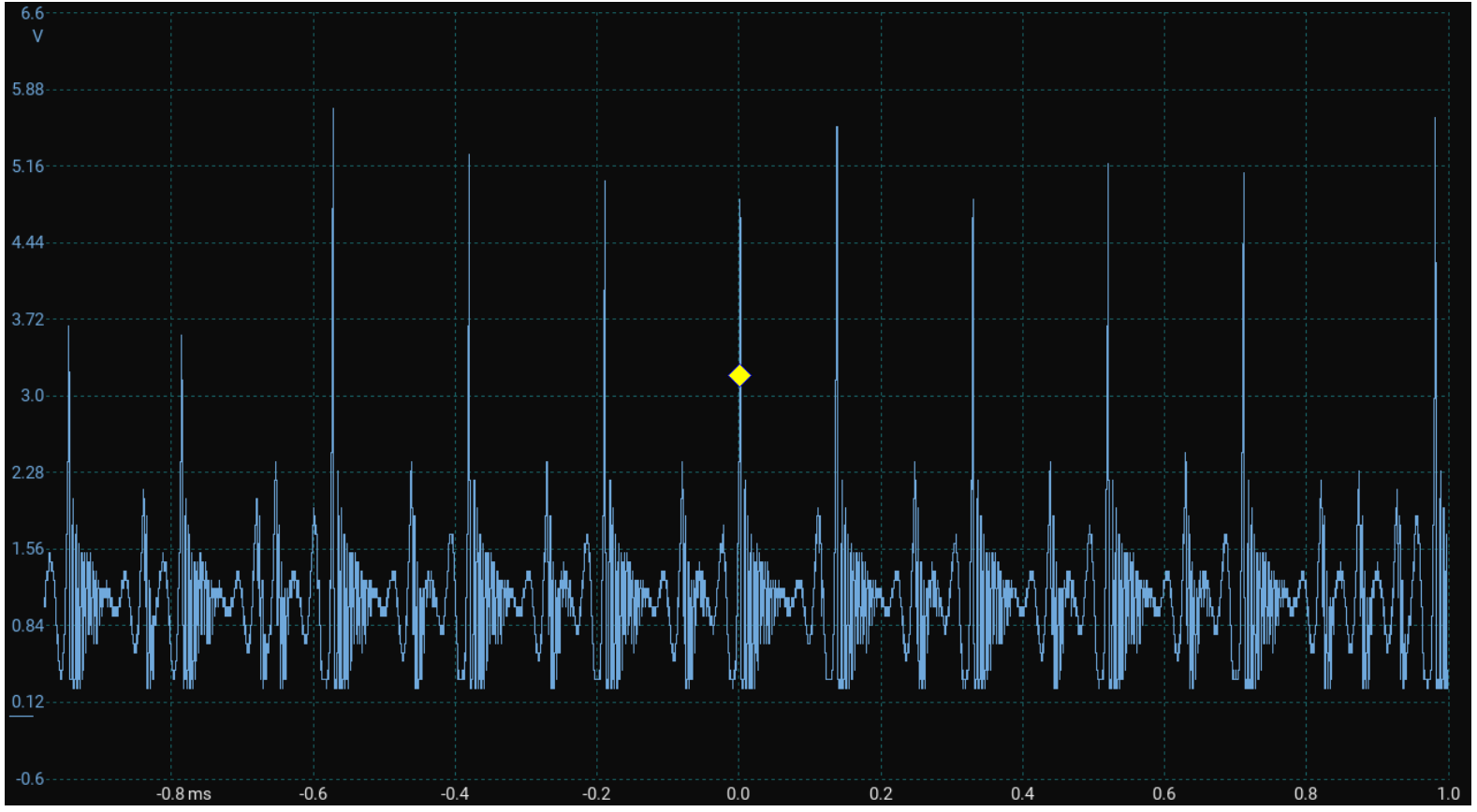
So definitely not periodic yet.



The difference of the most correlated 0.5 ms piece and the last one

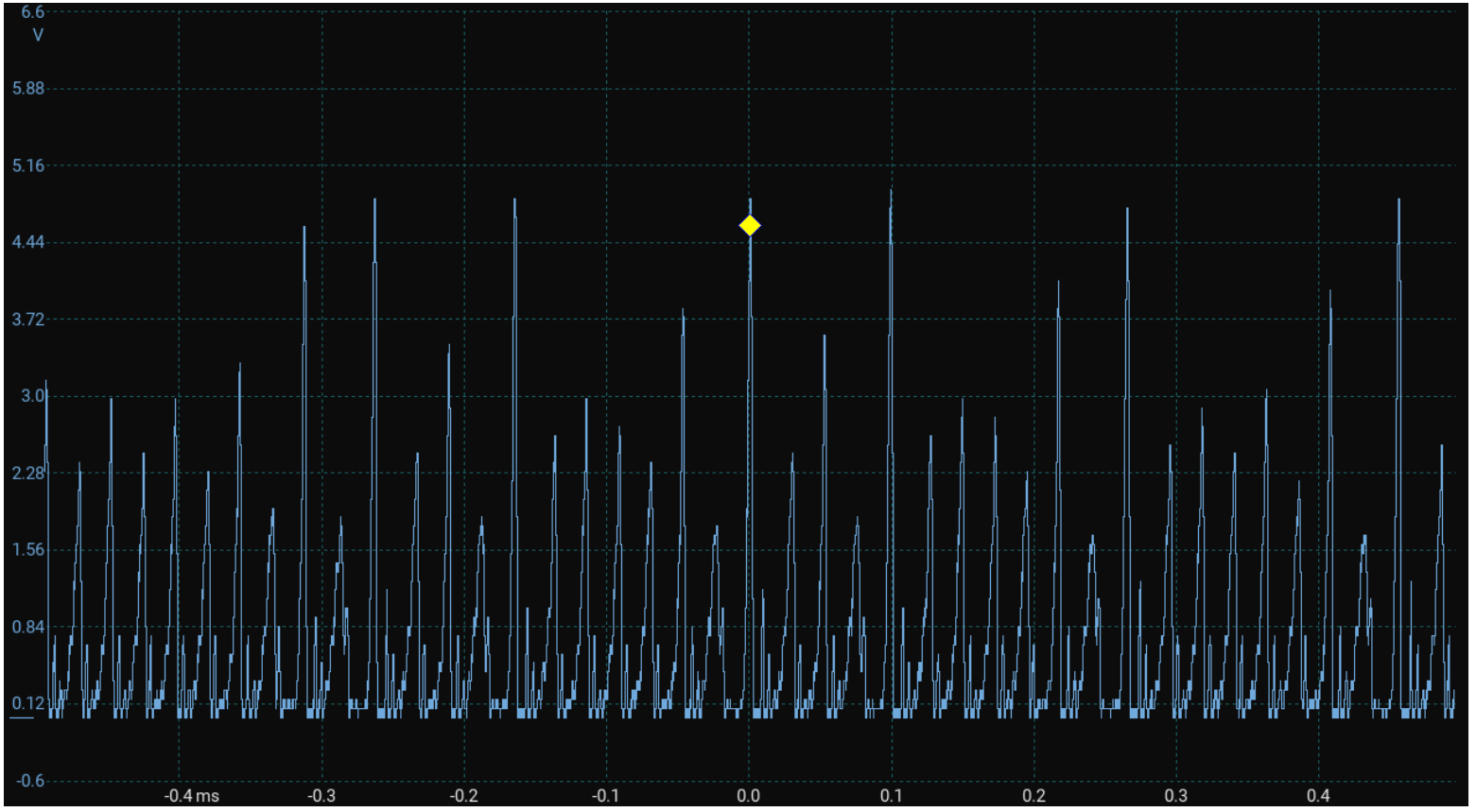


Experimental result I of the spiking oscillator



- Capacitor changed to 2200pF
- Resistor 1.68 kΩ

Experimental result II of the spiking oscillator



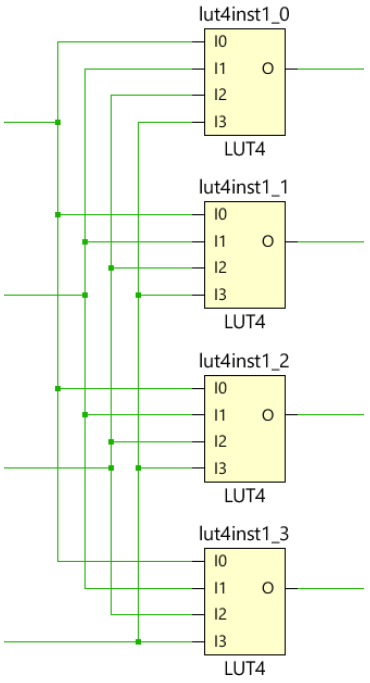
- Capacitor 2200pF
- Resistor 15.7 kΩ

What can run chaotically (?) on FPGAs?

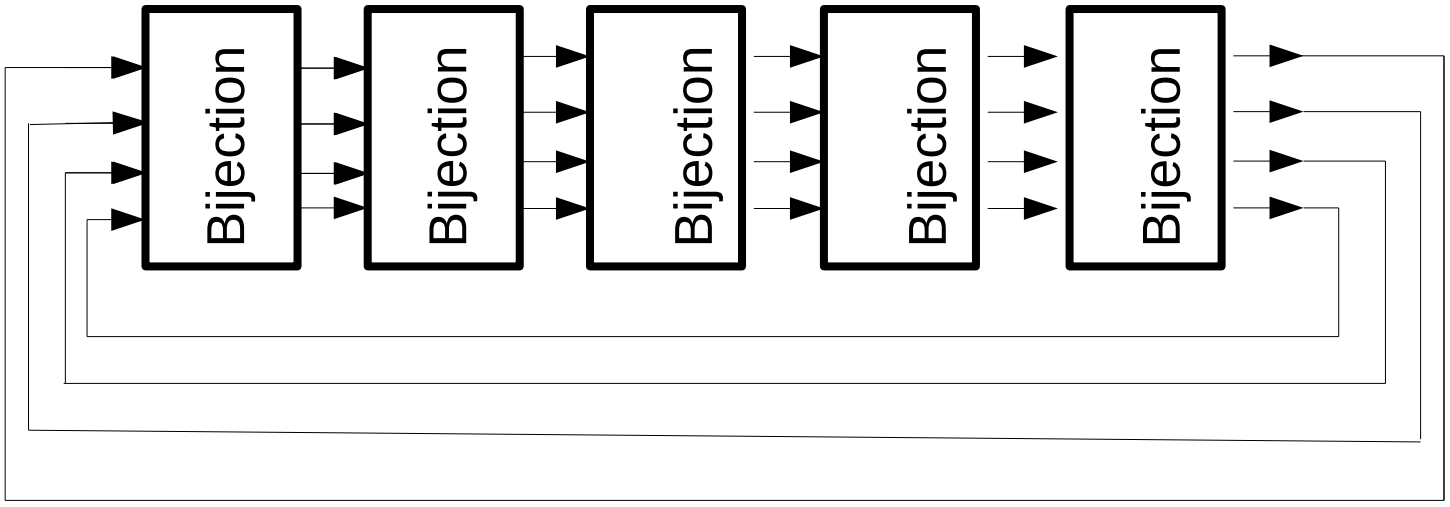
- FIROs and GAROs
- Combinatorial Multipliers with feedback (invented by Xilinx, not by me as wrongly claimed at Cryptarchi 2013)
- Combinatorial implementations of block ciphers with feedback
- Multitrack ring oscillators

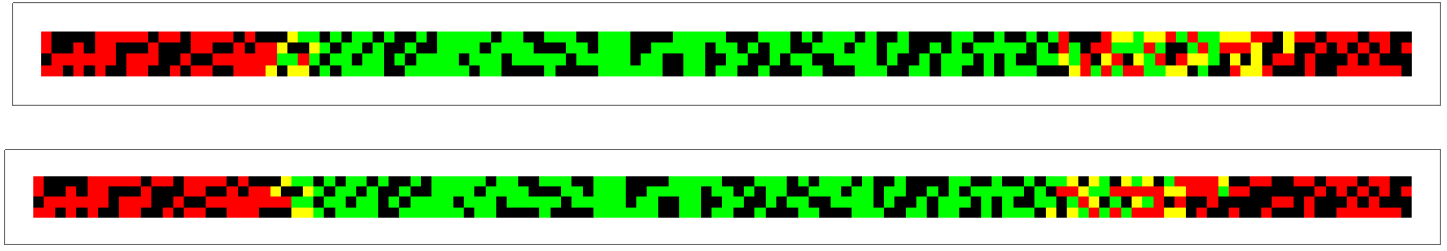
Problem: All of these can show periodic instead of chaotic oscillations

Multitrack ring oscillators



One 4-track bijection on an FPGA





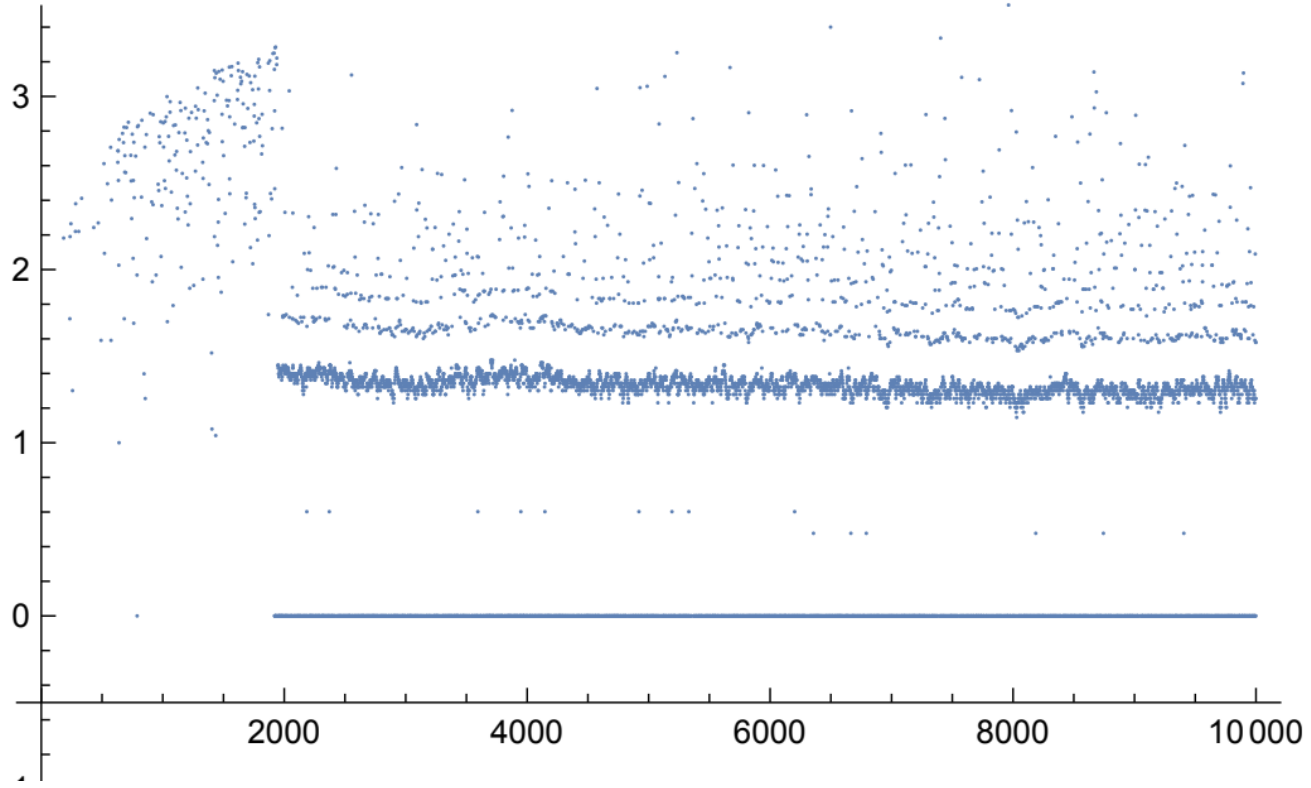
Off topic: no ring needed, so no oscillation

Indications of failing chaos

- Dichtl, Golic: High speed true random number generation with logic gates only, CHES 2007: We honestly described that there were some periodic oscillations for very short feedback polynomials
- Dichtl: Fibonacci Ring Oscillators as True Random Number Generators - A Security Risk, IACR Preprint 2015, also Cryptarchi 2017
- Shuqin Su, Bohan Yang, Vladimir Rožić, Mingyuan Yang, Min Zhu, Shaojun Wei, Leibo Liu: A Closer Look at the Chaotic Ring Oscillators based TRNG Design, IACR Preprint 2023/40

Full state sampling

- Introduced in my 2015 paper, samples all parts of a TRNG simultaneously and periodically
- How many samples ago have we seen this state for the last time?



The Su et al. paper: mixed blessings

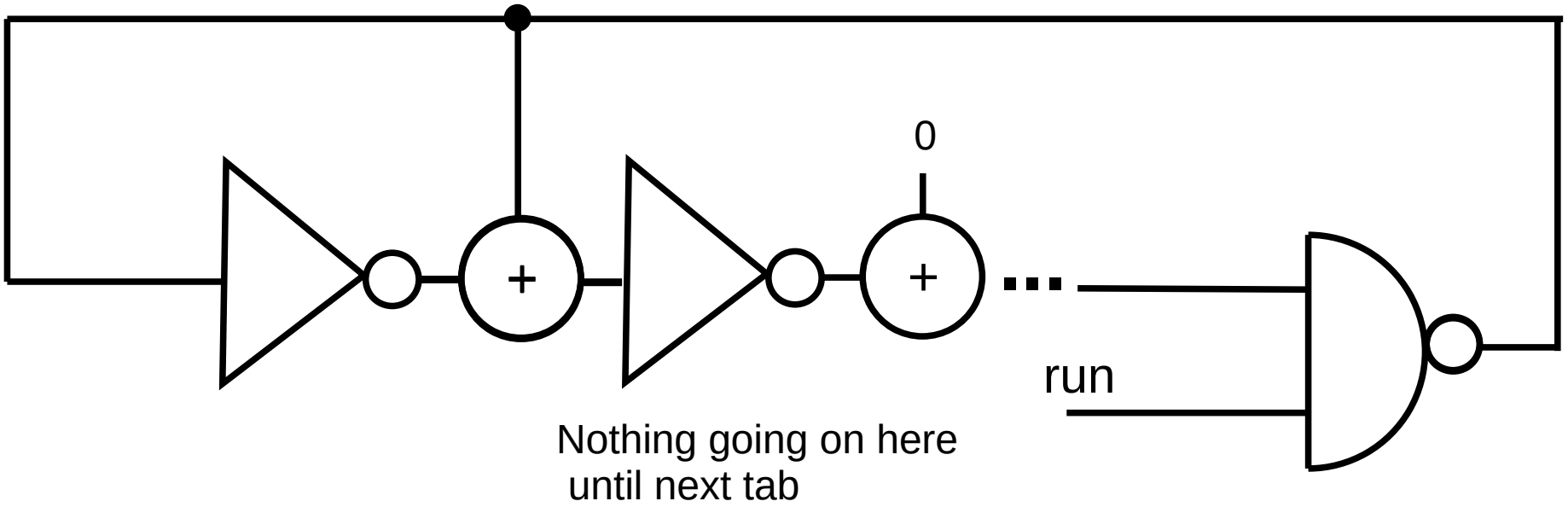
- Great improvements over my 2015 paper in many aspects
- Provides an explanation for failing chaos
- Suggests model to select good configurations
- Transferring GARO measurements from FPGA to CMOS design based on the fact both are 28nm technology
- Lowlight: Determining the Lyapunov exponent by measuring how fast a bit value increases from 0 to 1

Tricks learnt from the the Su et al. paper

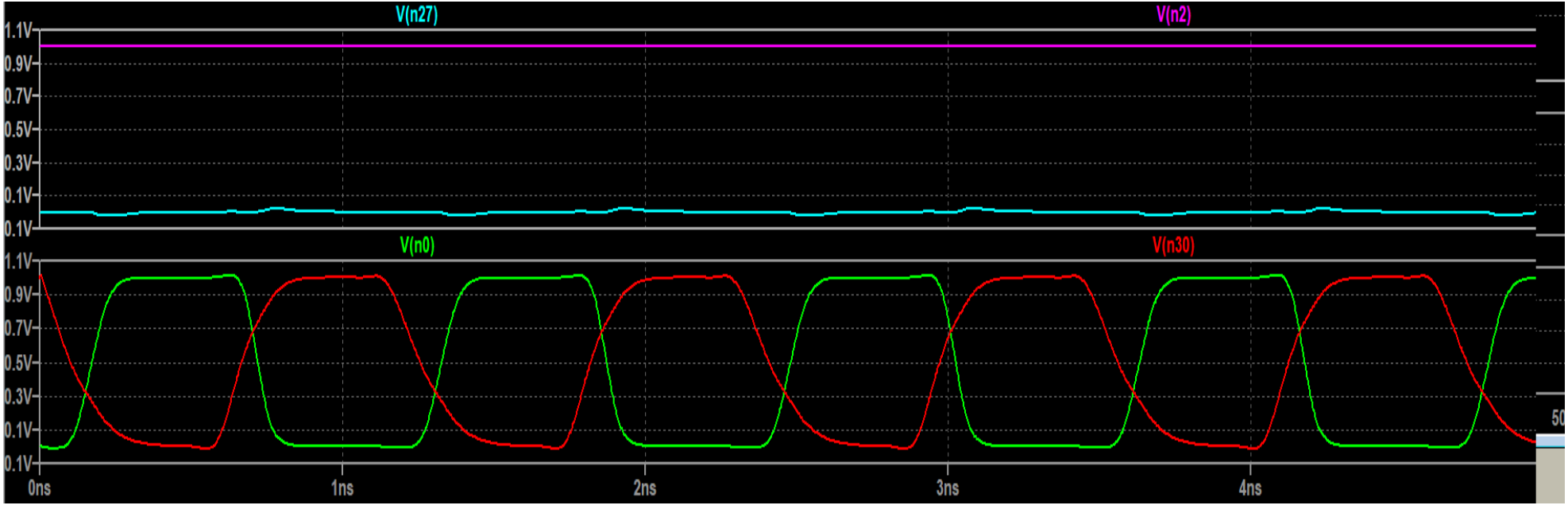
- Implementing also XORs not needed in a GARO makes the design homogeneous (is this good or bad?)
- Introducing delay elements with variable transistor sizes gives the designer an additional degree of freedom (alas not for FPGAs)

Almost killing a GARO I

- Learnt from Su et al.
- GARO of length 31: $x^{31}+x^{30}+x^{3}+1$



Almost killing a GARO II



Stochastic models for chaotic TRNGs

- Of course realistic stochastic models in the sense of AIS 31 make a lot of sense!
- But for which TRNGs (especially on FPGAs) have adequate stochastic models been achieved?
- In my opinion, it seems rather hopeless to provide adequate stochastic models for the rather complicated chaotic TRNGs.
- But wait...

The Bucci Luzzi chaotic TRNG

- Marco Bucci, Raimono Luzzi, A Fully-Digital Chaos_Based Random Bit Generator, in The New Codebreakers, Editors: Ryan, Naccache, Quisquater, Springer Lecture Notes in Computer Science, 2016, also Cryptarchi 2016
- Very simple design, just two ring oscillators:
  One with constant frequency, one with two switchable frequencies, switches to fast at the end of its own period and switches to slow at the end of the period of the other RO.
- Easy to see that it is indeed chaotic
- If there is a good stochastic model for ROs and if one can cope with RO interactions, it should be possible to give a good model for the Bucci Luzzi TRNG
- Problem for FPGA implementation: One must keep phase when switching occurs. Easily implemented on an ASIC by switching driver strength.
- Problem on FPGA (or at least on my Artix board): there is no internal tristate which could be used to drive a signal with various strengths. But there is tristate for output...