Impact of the Flicker Noise on the Ring Oscillator-based TRNGs

Licinius Benea*, Florian Pebay-Peyroula*, Viktor Fischer***, Romain Wacquez**, Mikaël Carmona*

* Univ. Grenoble Alpes, CEA, Leti,Grenoble, France

** CEA-Leti, Mines Saint-EtienneGardanne, France

*** Faculty of Information Technologies, Czech Technical University, Prague, Czech Republic

This work is supported by the “France 2030” government investment plan managed by the French National Research Agency (ANR), under the project ARSENE (ANR-22-PECY0004)

Outline

Introduction

- True Random Number Generators (TRNG): the basic building block of most cryptographic system
- Also used in :
  - Simulation
  - AI
  - Gambling
- Contrary to DRNG (Deterministic), they use a real physical noise source
- Principles:
  - Jitter – ring oscillators, PLL, STR – this presentation
  - Metastability
  - Chaos

How random is random?

Example 1 : Which is more random?

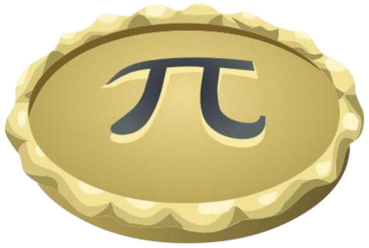
1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

Equiprobable

Intuitively, streaks are not random, but 50,6% of 20 random bits have streaks of ≥5

Example 2 : Is this random?

0 0 1 0 0 1 0 0 0 0 1 1 1 1 1 1 0 1 1 0 1 0



Random, but can be guessed with the right knowledge

Proving randomness

- Looking at generated random numbers does not fully guarantee randomness
- Statistical tests have a non-zero probability of suffering from type I ot type II errors (false positive or false negative)
- Standards (AIS 20/31) require a stochastic model to prove randomness

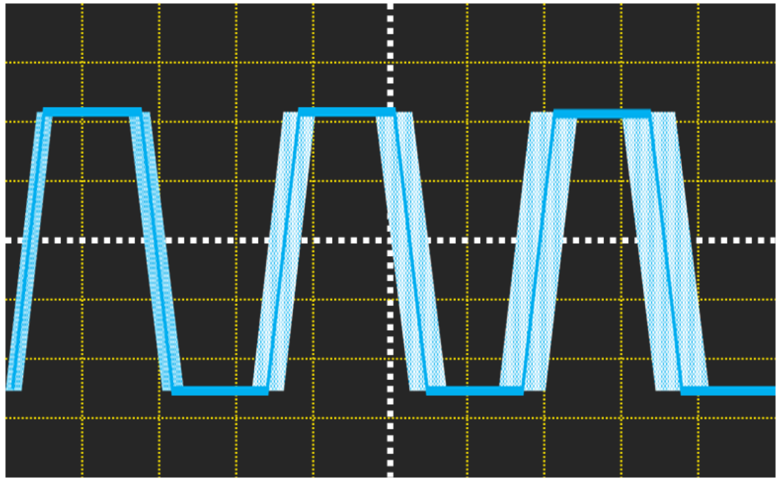
Use of noise models (phase noise)

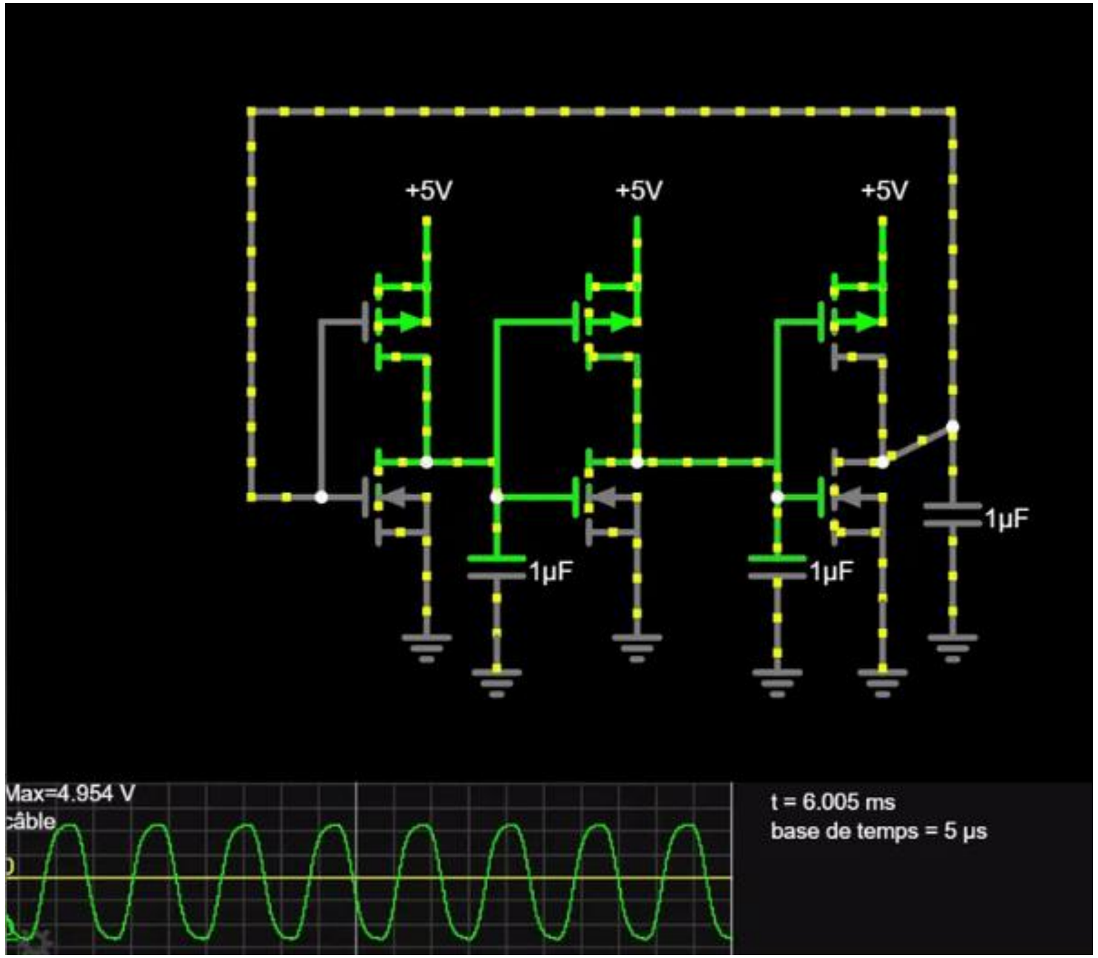
Identification of the physical phenomena causing entropy → Generate a stochastic model (TRNG)

2. Noise sources and TRNG structures

Ring oscillator description

- Structure: odd number of inverters
- The periodical signal is not perfect - jitter
- Jitter increases with accumulation time





Physical noise sources

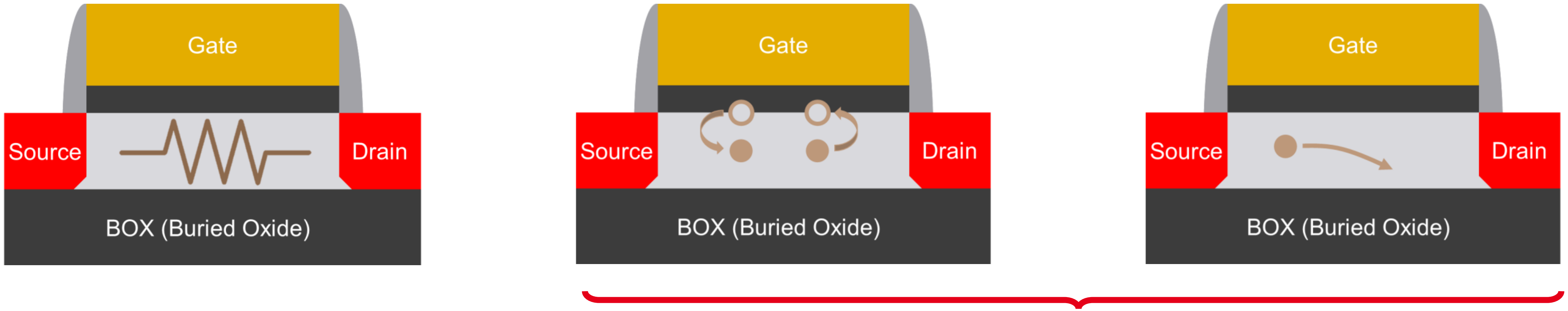
Thermal noise

[Nyquist 1928]

Flicker noise

Carrier Number Fluctuation (CNF)
[McWorther 1957]

Carrier mobility flucturations (CMF)
[Hooge 1969]



Random, not correlated

Random, correlated

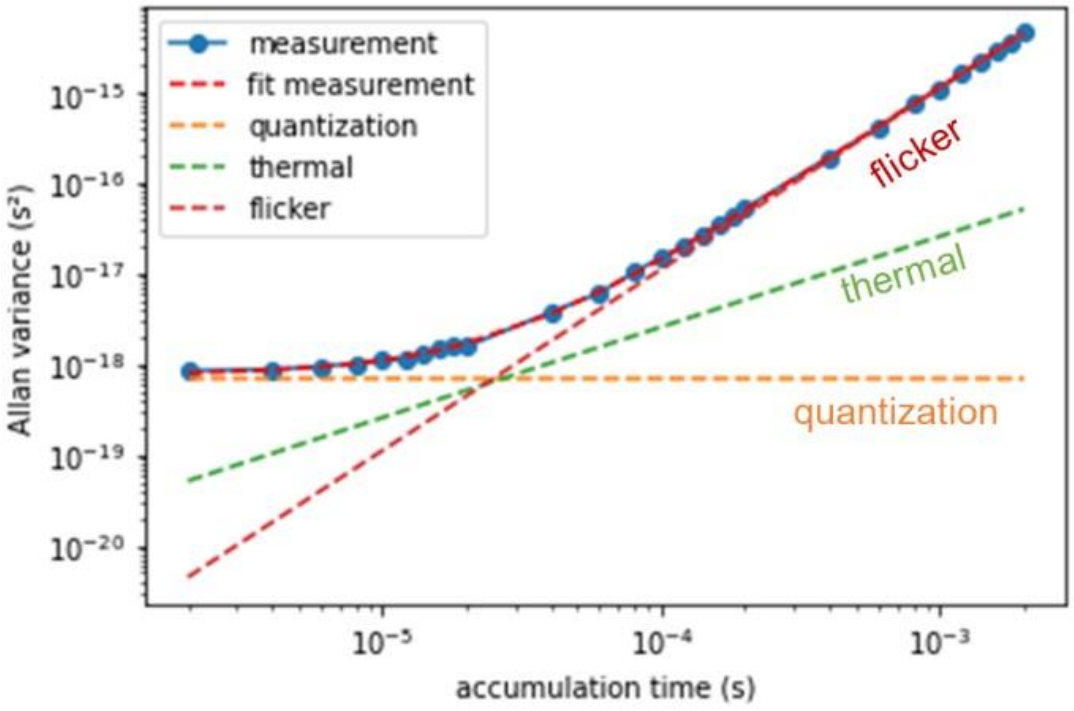
For TRNGs : autocorrelation = predictability

Sources of jitter (non-exhaustive)

- Global
- Deterministic
- Parasitic

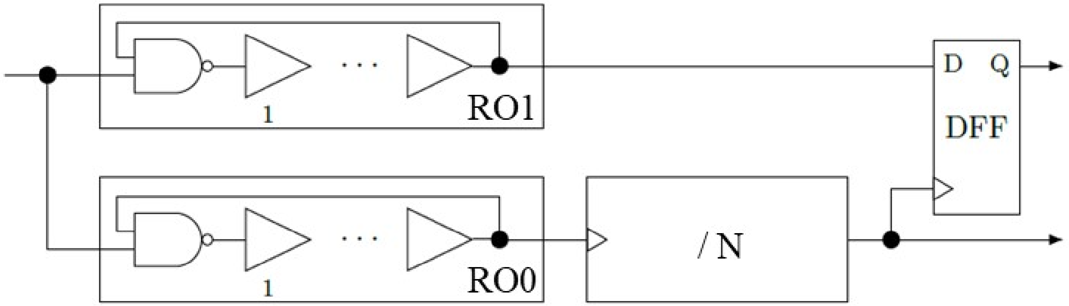
reduced by the differential principle

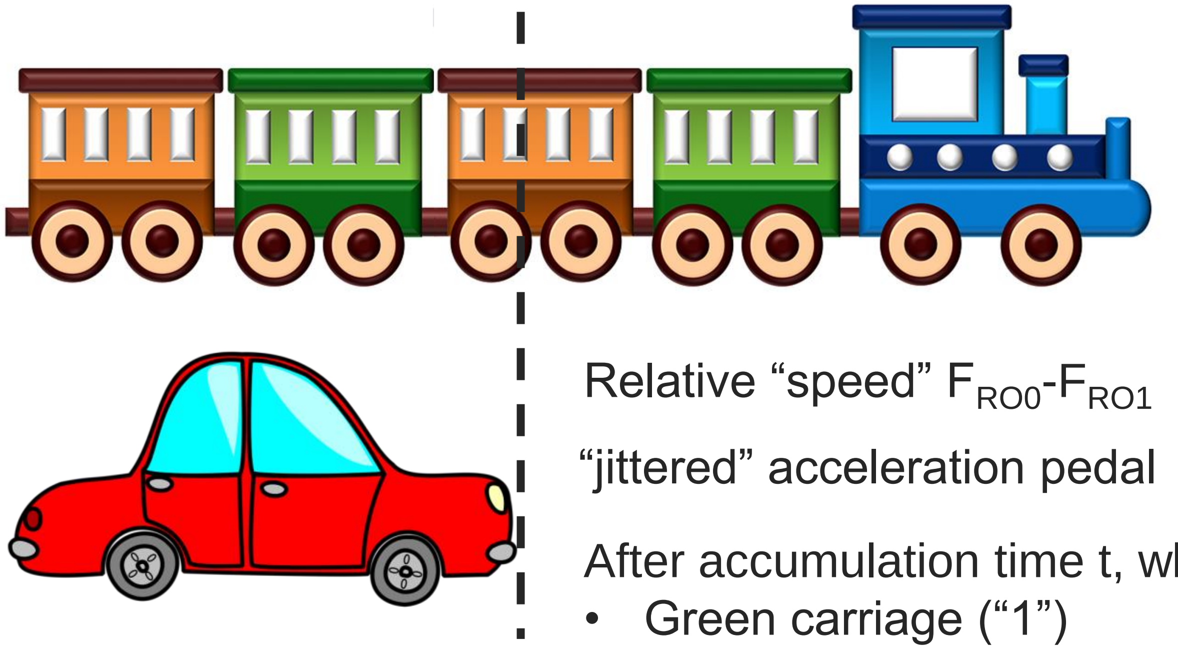
- Cross-talk → isolation
- Measurement
  - Quantization
- Physical
  - Thermal
  - Flicker



Use of jitter in TRNGs

- Elementary RO TRNG:
  - RO0 as a reference clock generator
  - RO1 as an entropy source





Relative “speed” F_{RO0}-F_{RO1}

“jittered” acceleration pedal

After accumulation time t, what is the position of the car?

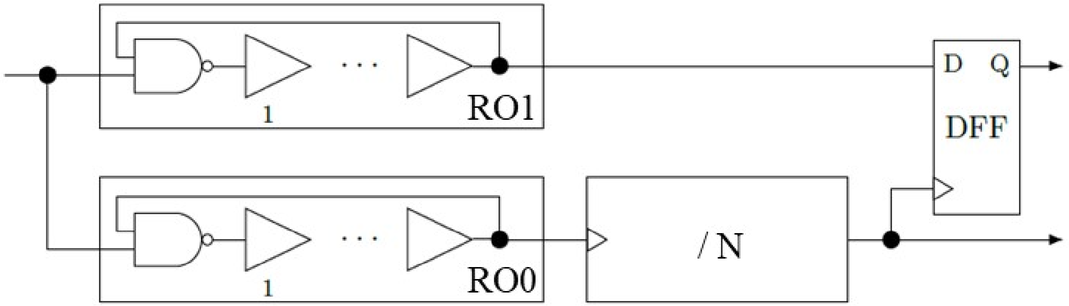
- Green carriage (“1”)
- Brown carriage (“0”)

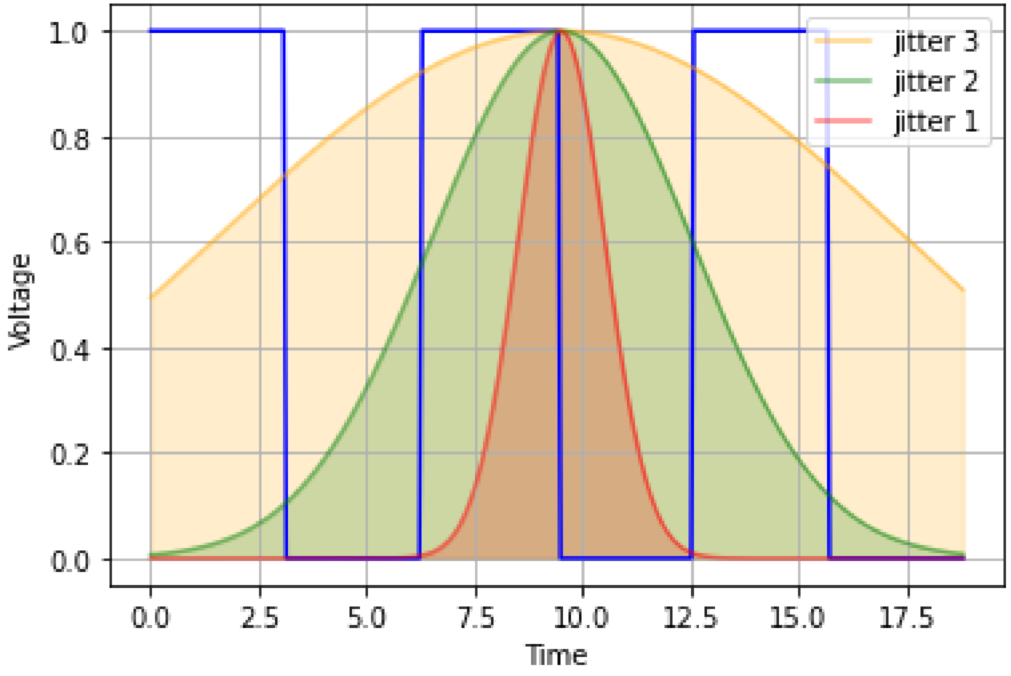
Use of jitter in TRNGs

- Elementary RO TRNG:
  - RO0 as a reference clock generator
  - RO1 as an entropy source

- Principle of jitter transfer – referential change
  - 1 perfect RO signal (clk1)
  - 1 jittered RO signal (clk0)

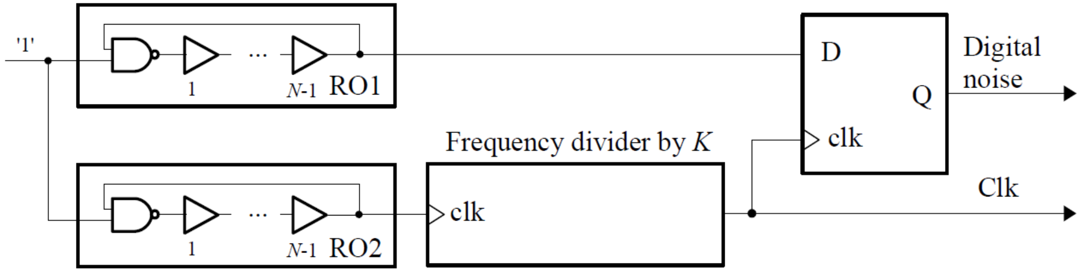
- Frequency divider: accumulated jitter is large enough for the entropy requirements





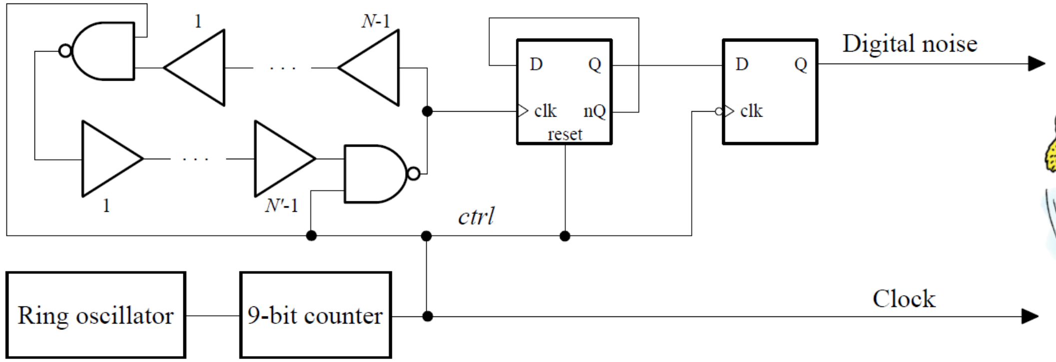
Other types de RO-TRNG

1. Elementary RO-TRNG



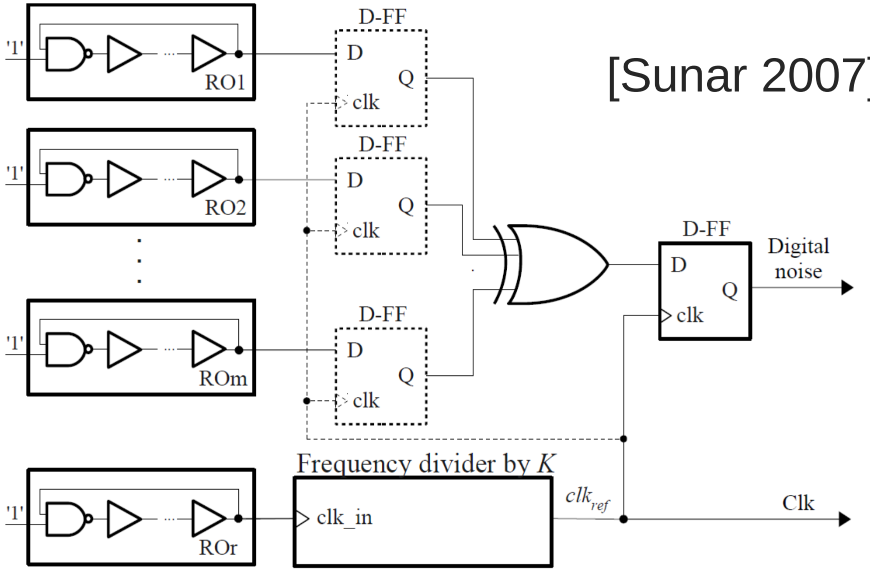
[Baudet 2011]

2. Transient Effect RO-TRNG

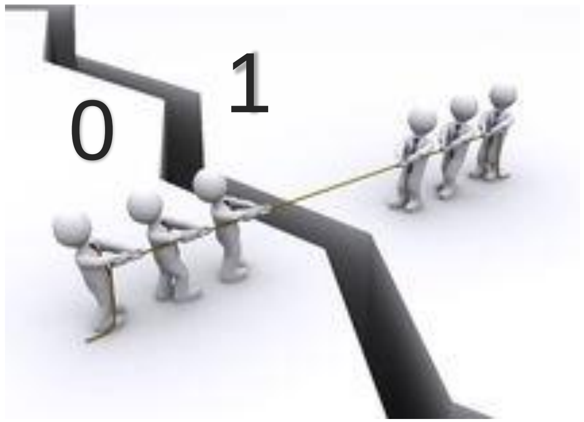


[Haddad 2015]

3. Multi-RO-TRNG



[Sunar 2007]

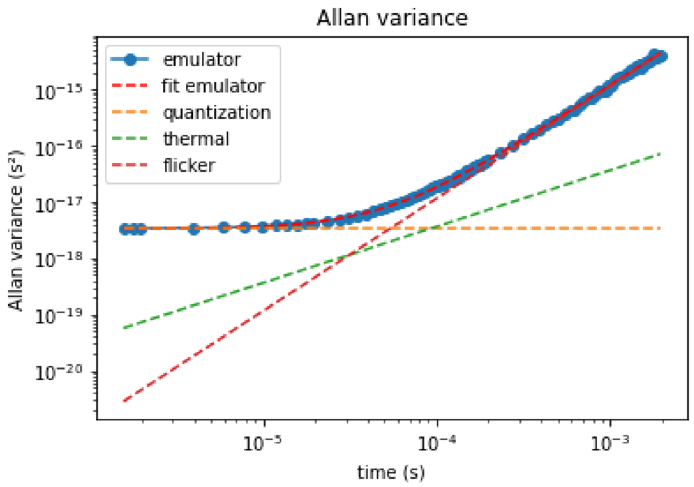


Current entropy models

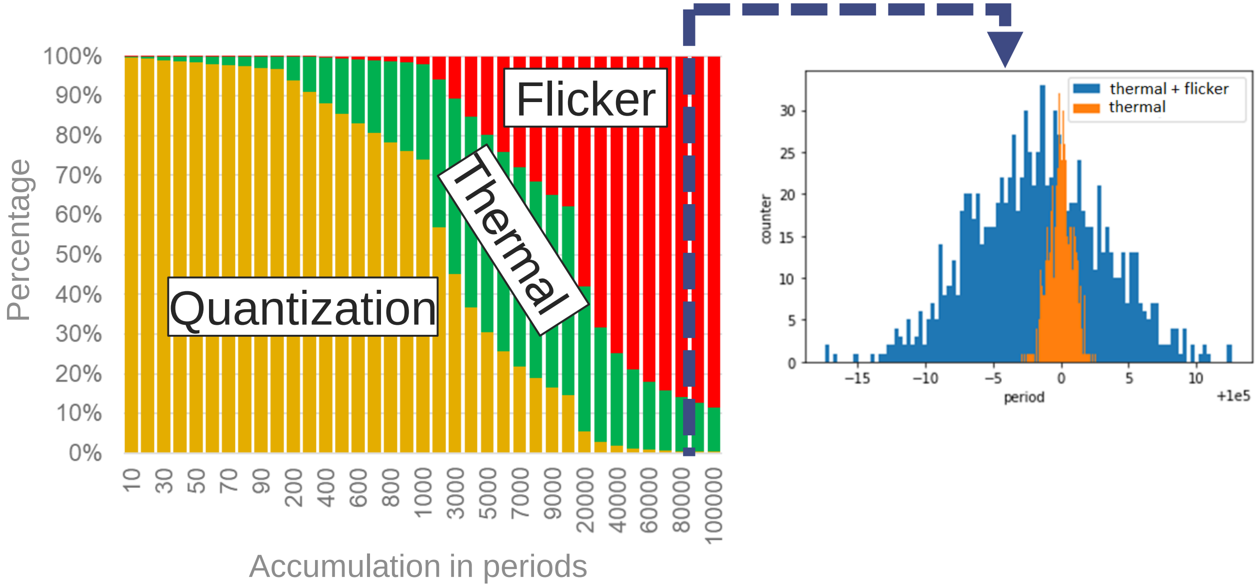
1. Isolate and use only the thermal jitter component
2. Postulate that only the thermal noise contributes to the entropy rate of the TRNG

Issues:

1. How can one be sure that thermal noise is well determined? (hidden by quantization)



2. TRNG working point is in a flicker dominated region. Influence on entropy?



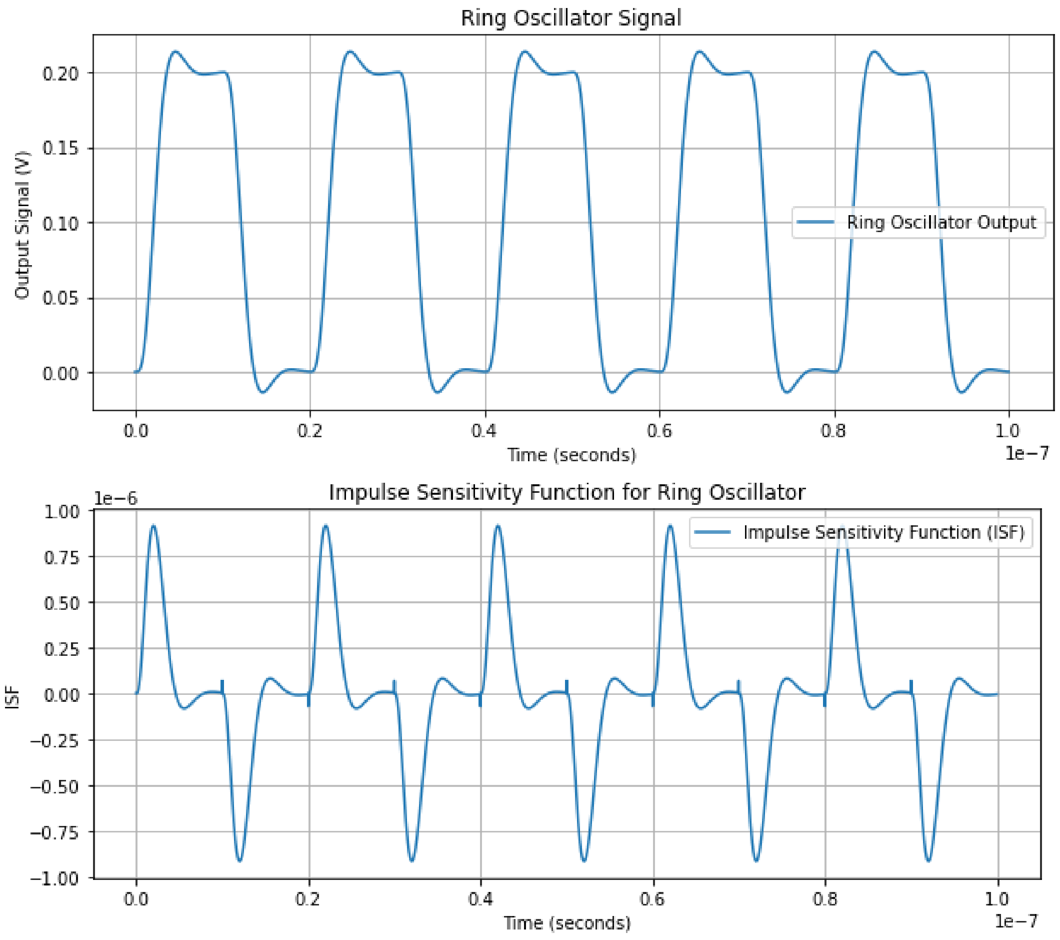
Solution : Emulator

3. Emulator

Motivation and principle

- Motivation:
  - We need to modify the amplitudes / to cancel noise sources on demand
  - Conventional simulation tools may take ~week to simulate 1M periods of a RO
- Use of the [Hajimiri 1999] model :
  - There is an impact of the noise on phase noise only during transient phases
  - Susceptibility of a signal to be influences in terms of phase noise (Impulse Sensitivity Function)
- Absolute phase :

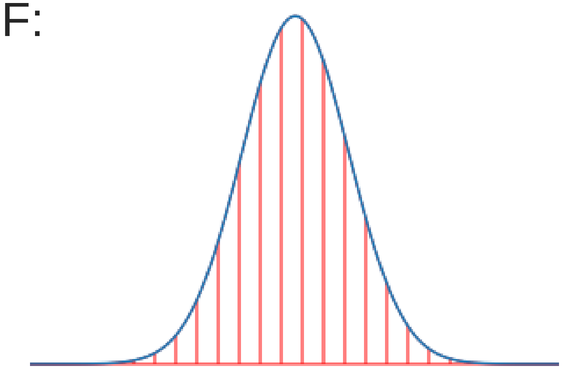
$$\phi(t) = \int_{-\infty}^{t} \frac{\Gamma(\omega_0 \tau)}{q_{max}} \cdot i(\tau) \cdot d\tau$$



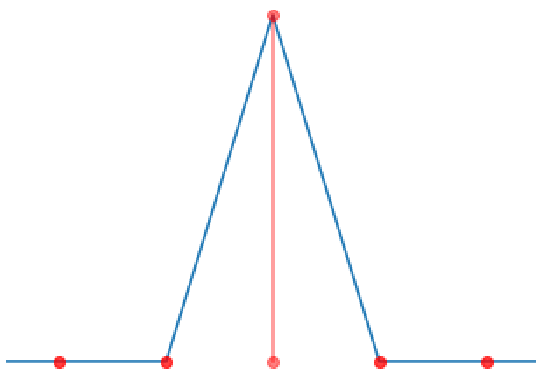
*schematic

Hypothesis for emulator

- Simplification : TRNGs only need the trigger moments of rising (/falling) edges
- ISF:



$\Rightarrow$



$\Leftrightarrow$

reduction

$$\frac{\Gamma(\omega_0\tau)}{q_{max}} \cdot i(\tau) \ = \ A \cdot \delta i$$

$A \ - all\ encompasing\ amplitude\ term$

$\delta i \ - generic$ "unitary" $noise\ term$

- Time deviation :

$i_{th}(\tau)$ - thermal

$i_{fl}(\tau)$ - flicker

$$\frac{2\pi}{\omega} \cdot \phi(t) = \frac{2\pi}{\omega} \cdot \int \frac{\Gamma(\omega_0\tau)}{q_{max}} \cdot i(\tau) \cdot d\tau \xrightarrow{assembling} \frac{2\pi}{\omega} \cdot d\phi(t_i) = A_{th} \cdot \delta i_{th}(t_i) + A_{fl} \cdot \delta i_{fl}(t_i)$$

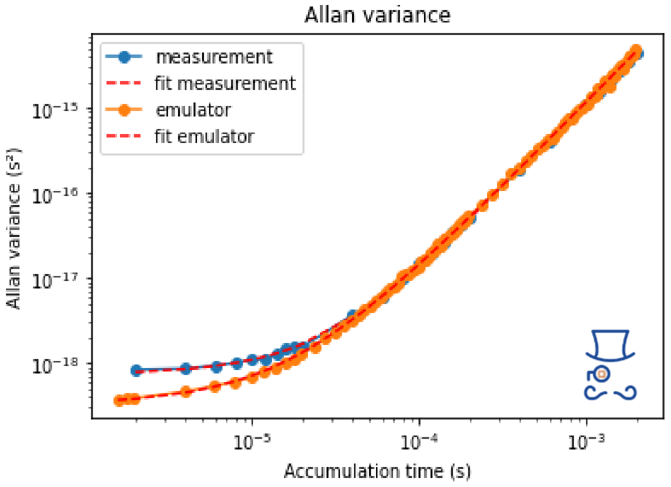
- Each period of the RO : $\mathrm{d}t_i = T_0 + A_{th} \cdot \delta i_{th}(t_i) + A_{fl} \cdot \delta i_{fl}(t_i)$
- Rising edges : $t_i = i \cdot T_0 + \sum_{-\infty}^{t_i} \left(A_{th} \cdot \delta i_{th}(t_i) + A_{fl} \cdot \delta i_{fl}(t_i)\right)$

A_{th}, A_{fl} are amplitudes $\rightarrow$ to be calibrated
$\delta i_{th}, \delta i_{fl}$ are generic terms for thermal and flicker noises (Python colorednoise)

Validation

ASIC (28nm FD-SOI, 101 elements, 500MHz)

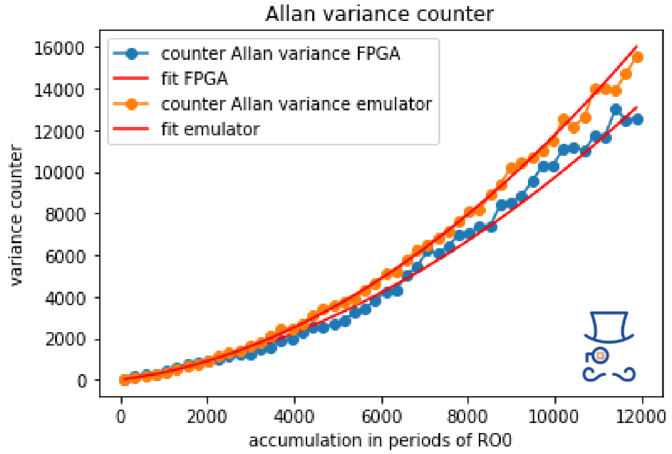
$$dt_i^{RO} = T_0 + \sqrt{\frac{a_1 \cdot T_0}{factor_{th}}} \cdot \delta i_{th}^{RO} + \sqrt{\frac{a_2 \cdot T_0^{\,2}}{factor_{fl}}} \cdot \delta i_{fl}^{RO}$$



	a_2 (flicker)	a_1 (thermal)	a_0 (quantization)
Measurement	$1{,}11 \cdot 10^{-9}$	$2{,}56 \cdot 10^{-14}$	$7{,}37 \cdot 10^{-19}$
Emulator	$1{,}16 \cdot 10^{-9}$	$2{,}81 \cdot 10^{-14}$	$3{,}23 \cdot 10^{-19}$
Error (%)	4,75%	9,75%	56,21%

FPGA(ARTY A7)

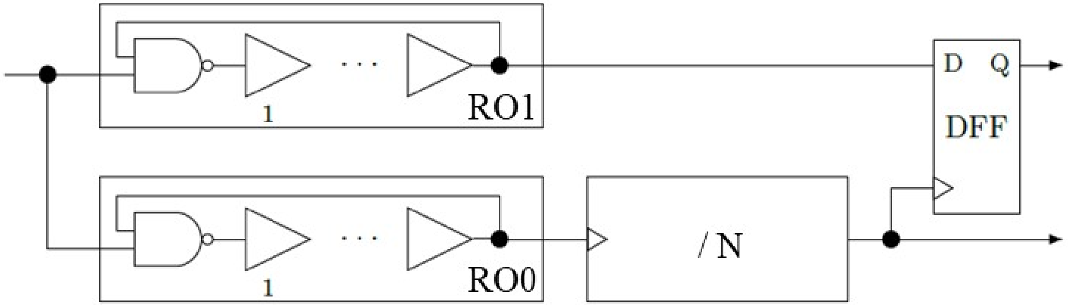
$$dN_i = N + \sqrt{\frac{a_1 \cdot N}{factor_{th}}} \cdot \delta i_{th}^{RO} + \sqrt{\frac{a_2 \cdot N^2}{factor_{fl}}} \cdot \delta i_{fl}^{RO}$$



	a_2 (flicker)	a_1 (thermal)	a_0 (quantization)
Measurement	$6{,}90 \cdot 10^{-5}$	$2{,}81 \cdot 10^{-1}$	$1{,}15 \cdot 10^{1}$
Emulator	$9{,}13 \cdot 10^{-5}$	$2{,}62 \cdot 10^{-1}$	$9{,}51 \cdot 10^{1}$
Error (%)	32,3%	6,72%	91,8%

Emulating an Elementary RO TRNG

- Hypothesis:
  - We “transfer” all phase noise into one of the ROs
- $$\begin{cases} dt_i{}^{RO1} = T_0{}^{RO1} \\ dt_i{}^{RO0} = T_0{}^{RO0} + A_{th} \cdot \delta i_{th}{}^{RO0} + A_{fl} \cdot \delta i_{fl}{}^{RO0} \end{cases}$$
- If duty cycle = 50%:



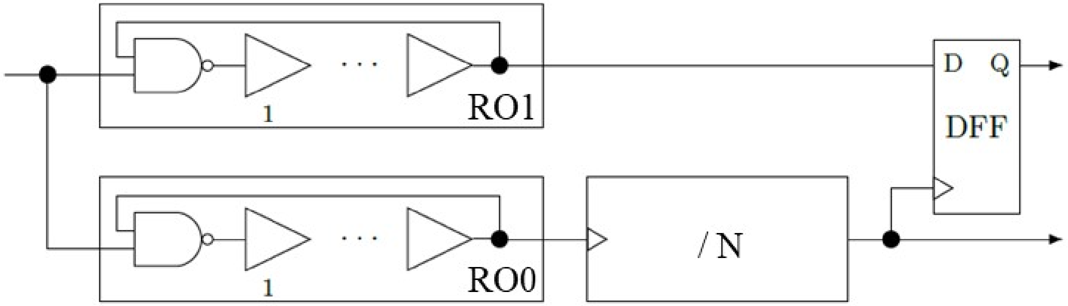
Output bit : $\left\lfloor \frac{absolute\ jitter\ of\ RO_0}{T_0{}^{RO1}}\ mod\ 1 + 0.5 \right\rfloor$

$$\left\lfloor \frac{\sum_{t_i}\left(N \cdot T_0{}^{RO0} + \sqrt{\frac{a_1 \cdot N \cdot T_0{}^{RO0}}{factor_{th}}} \cdot \delta i_{th}{}^{RO0}(t_i) + \sqrt{\frac{a_2 \cdot N^2 \cdot T_0{}^{RO0^2}}{factor_{fl}}} \cdot \delta i_{fl}{}^{RO0}(t_i)\right)}{T_0{}^{RO1}} mod 1 + 0.5 \right\rfloor$$

Emulating an Elementary RO TRNG (Python)

- Python script uses colorednoise library:
  - Based on [Timmer 1995]
  - For each frequency of the spectrum we generate random sequences whose amplitude is proportional to the desired spectrum

$$S(\omega) \sim (1/\omega)^{\beta}$$




```python
def ERO_bits(T1,T2,Ath,Afl,N,size):

    #generate noise
    di_thermique = cn.powerlaw_psd_gaussian(0 ,size)
    di_flicker = cn.powerlaw_psd_gaussian(1,size)

    dti_emulator=N*T2*np.ones(size)
           +di_thermique*np.sqrt(Ath*T2*N/factor_th)
           +di_flicker*np.sqrt(Afl*((N*T2)**2)/factor_fl)

    ti_emulator=np.cumsum(dti_emulator)
    bits=np.round((ti_emulator/T1)%1)
```

Script can be found at:

Free

OpenTRNG

https://opentrng.org/

https://github.com/opentrng/papers/tree/master/ches2024

4. Consequences of noise (flicker) on random number generation

Autocorrelation of bits

- Autocorrelation represents the predictibility introduced by flicker noise

$$\rho_k = \sum_i \frac{(X_{i+k} - E(X_{i+k}))(X_i - E(X_i))}{\sigma_{X_{i+k}} \cdot \sigma_{X_i}}$$

- Variation of flicker noise amplitude of the time series:

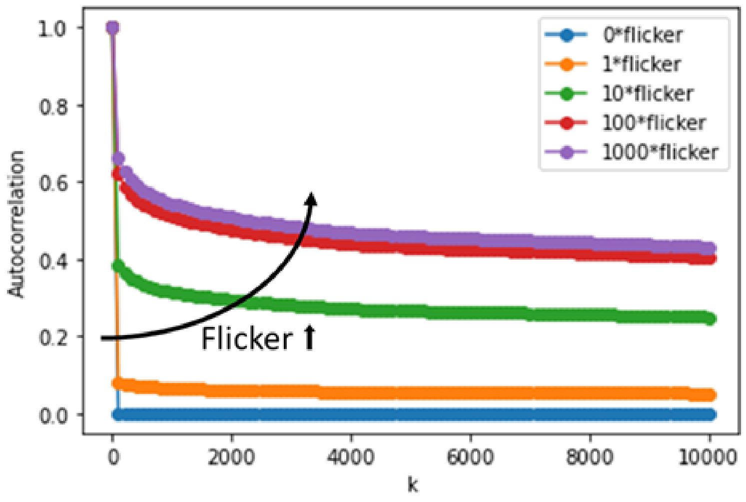
$$dt_i{}^{RO} = N \cdot T_0 + \sqrt{\frac{a_1 \cdot N \cdot T_0}{factor_{th}}} \cdot \delta i_{th}{}^{RO} + \sqrt{\frac{M \cdot a_2 \cdot (N \cdot T_0)^2}{factor_{fl}}} \cdot \delta i_{fl}{}^{RO}$$

- Bit series:

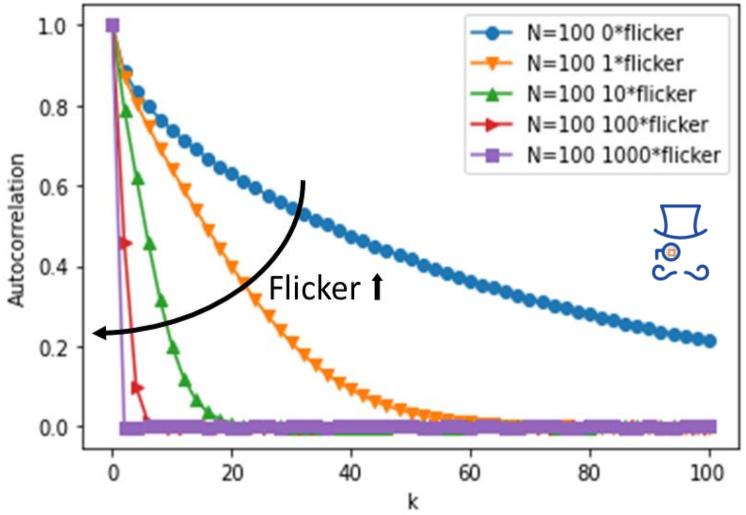
$$\left\lfloor \frac{t_i{}^{RO0}}{T_0{}^{RO1}} mod1 + 0.5 \right\rfloor$$

- Sampling might reduce the autocorrelation effect introduced by flicker noise

Raw time series

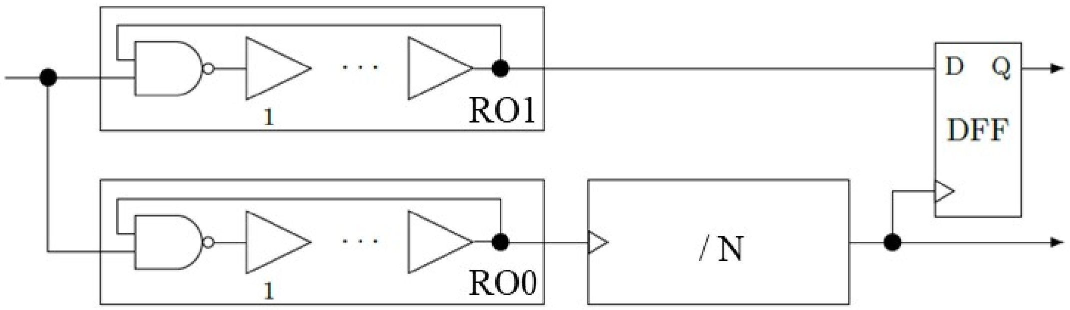


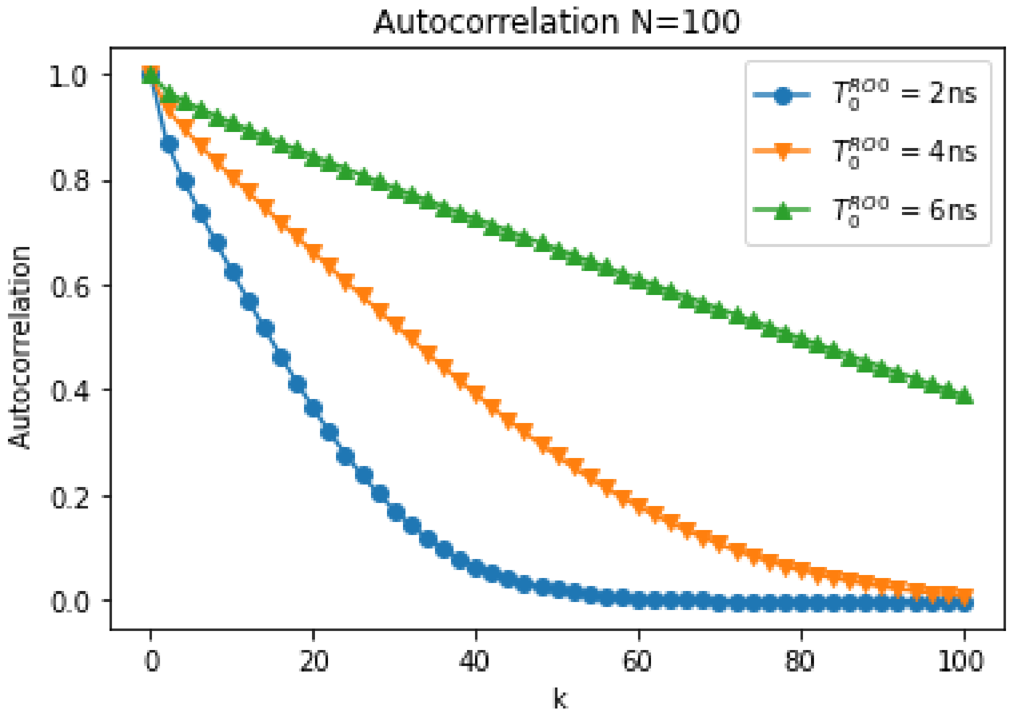
Bits of the ERO-TRNG



Influence of sampling on autocorrelation

- Assumption: « The sampling limits the autocorrelation introduced by flicker noise »
- If sampling with RO0 limits the depth of the autocorrelation ⇒ by changing the period of RO0, the depth of autocorrelation increases
- Conclusion: The sampling limits autocorrelation by a lower bound of the frequency

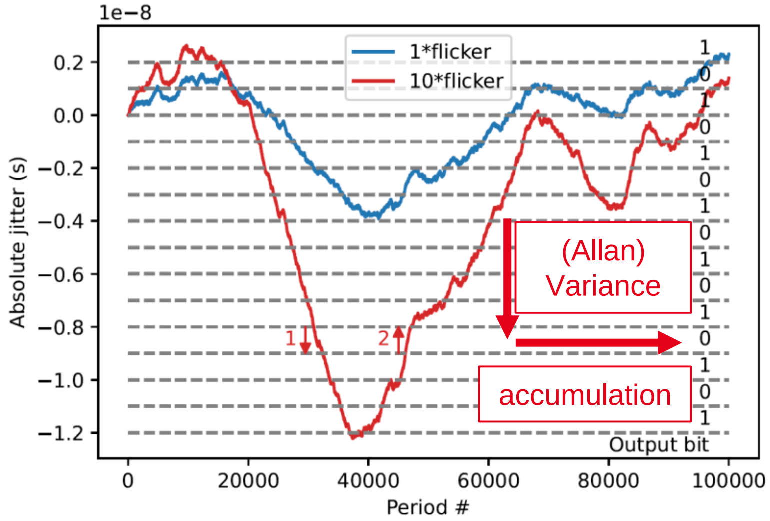


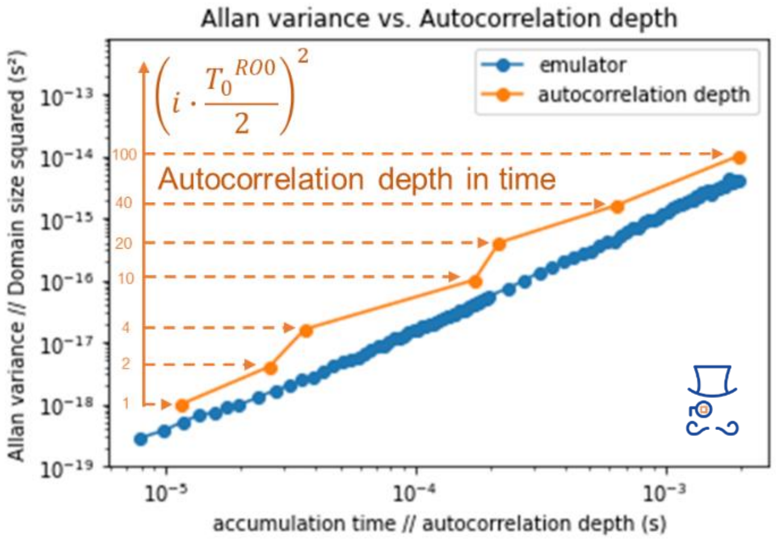


Depth of the autocorrelation

- Phase perspective : determination of the output bit
- Bit perspective: a transition (from “0” to “1”) does not allow to determine if it is the result of an increase or decrease in absolute jitter
  - For each transition ⇒ reset of the perceived phase and removal of the memory effect
- The deviation of half a “domain” → mean deviation is given by the (Allan) variance
- The depth of the autocorrelation is given by the accumulation time necessary for the Allan variance to reach half the period of the sampling RO

$$a_2 \cdot t^2 + a_1 \cdot t = \left(\frac{T_0^{RO0}}{2}\right)^2 \Rightarrow t = \frac{-a_1 + \sqrt{{a_1}^2 - 2 \cdot a_2 \cdot {T_0^{RO0}}^2}}{2a_2}$$





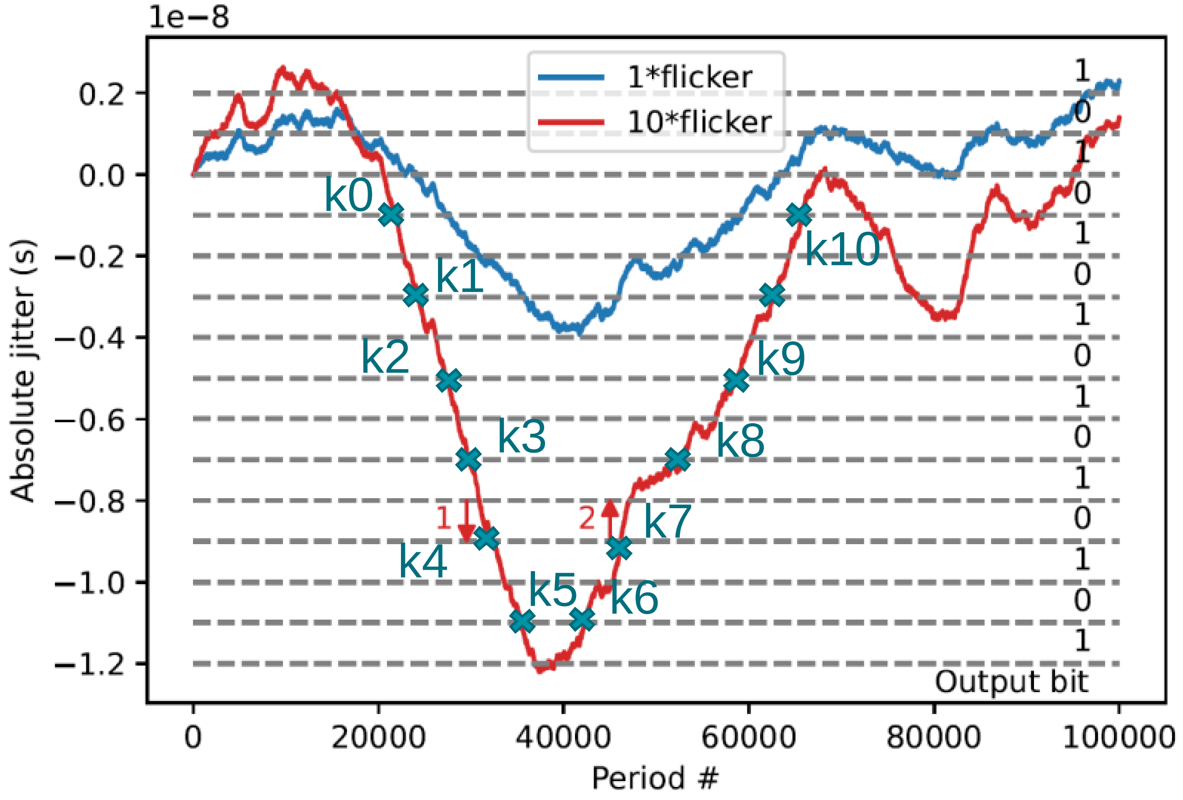
Jitter – bit relationship

- Absolute jitter:

$$\frac{2\pi}{\omega}\cdot\phi(t) = \frac{2\pi}{\omega}\cdot\int\frac{\Gamma(\omega_0\tau)}{q_{max}}\cdot i(\tau)\cdot d\tau\ \ (divergent)$$

- Output bit of ERO-TRNG (same T_0):

$$bit(t) = \left\lfloor \frac{\int_{-\infty}^{t}\frac{\Gamma(\omega_0\tau)}{q_{max}}\cdot i(\tau)\cdot d\tau}{2\omega_0 T_0}\ mod\ 1 + 0.5\right\rfloor$$



$$bit(t) = \left\lfloor \frac{\int_{t_0}^{t_1}\frac{\Gamma(\omega_0\tau)}{q_{max}}\cdot i(\tau)\cdot d\tau + \int_{t_1}^{t_2}\frac{\Gamma(\omega_0\tau)}{q_{max}}\cdot i(\tau)\cdot d\tau + \cdots + \int_{t_{k-1}}^{t_k}\frac{\Gamma(\omega_0\tau)}{q_{max}}\cdot i(\tau)\cdot d\tau + \int_{t_k}^{t}\frac{\Gamma(\omega_0\tau)}{q_{max}}\cdot i(\tau)\cdot d\tau}{2\omega_0 T_0}\ mod\ 1 + 0.5\right\rfloor$$

$$\frac{T_0}{T_0}\ mod\ 1 = 0$$

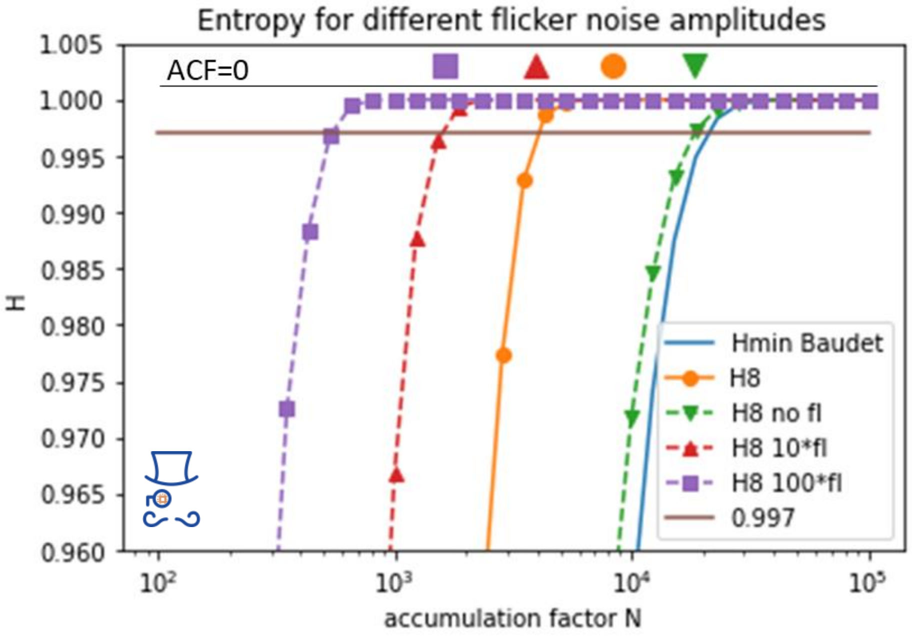
1: modulo function reinitialises phase

$$bit(t) = \left\lfloor \frac{\int_{t_k}^{t}\frac{\Gamma(\omega_0\tau)}{q_{max}}\cdot i(\tau)\cdot d\tau}{2\omega_0 T_0}\ mod\ 1 + 0.5\right\rfloor$$

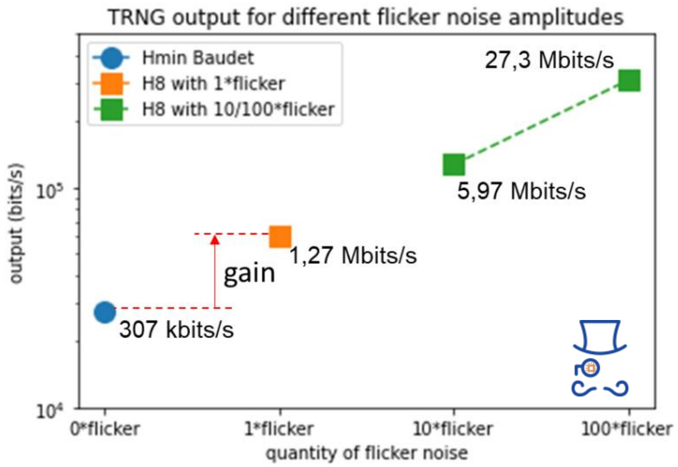
2: non-cancelled part remains in the same « domain »

The effect of the flicker noise on entropy

- Entropy rate calculated on 8 bits for different flicker noise amplitudes
- Orange curve : « standard » quantities of flicker/thermal noises
- Green curve with thermal component only fits [Baudet 2011] model
- Generally, flicker noise increases entropy



- Implications on output
  - For ACF = 0 and Entropy rate > 0.997 ⇒ 4x increase in the output for our device for “standard” noise quantities
  - Caution: those conditions are achieved at 99.98% flicker (i.e. 4x in output for 5000x flicker noise)



Conclusion

- Study of the influence of the flicker noise: emulator adapted to RO-TRNG applications
  - Tool: Simple emulator adapted to TRNG, Python-based
  - Advantage: Reproduction and/or modification of real parameters
- Influence of the flicker noise on the behaviour of the TRNG
  - Take away: From a bit perspective, autocorrelation doesn’t have any influence starting from the point where jitter is greater than the half-period
    - Flicker is not always harmful
    - Can improve the bit rate of the TRNG
    - Can simplify jitter characterization
    - Opens the perspective for a new stochastic model integrating it
  - What the paper does not provide: Entropy computation knowing previous bits or phases
- Perspectives
  - Emulator: study of other configurable conditions (drift, aging, duty cycle etc.) and other TRNG structures
  - Noise sources: development of a new stochastic model integrating noise sources adapted to advanced technological nodes (thermal, flicker, RTN, etc.)

Bibliography

[Allini 2018] E. Noumon Allini, M. Skórski, O. Petura, F. Bernard, M. Laban, and V. Fischer, ‘Evaluation and Monitoring of Free Running Oscillators Serving as Source of Randomness’, IACR Transactions on Cryptographic Hardware and Embedded Systems, vol. Volume 2018, pp. 214-242 Pages, Aug. 2018, doi: 10.13154/TCHES.V2018.I3.214-242.

[Baudet 2011] M. Baudet, D. Lubicz, J. Micolod, and A. Tassiaux, ‘On the Security of Oscillator-Based Random Number Generators’, J Cryptol, vol. 24, no. 2, pp. 398–425, Apr. 2011, doi: 10.1007/s00145-010-9089-3.

[Benea 2022] L. Benea, M. Carmona, F. Pebay-Peyroula, and R. Wacquez, ‘On the Characterization of Jitter in Ring Oscillators using Allan variance for True Random Number Generator Applications’, in 2022 25th Euromicro Conference on Digital System Design (DSD), Maspalomas, Spain: IEEE, Aug. 2022, pp. 534–538. doi: 10.1109/DSD57027.2022.00077.

[Haddad 2014] P. Haddad, Y. Teglia, F. Bernard, and V. Fischer, ‘On the assumption of mutual independence of jitter realizations in P-TRNG stochastic models’, in Design, Automation & Test in Europe Conference & Exhibition (DATE), 2014, Dresden, Germany: IEEE Conference Publications, 2014, pp. 1–6. doi: 10.7873/DATE.2014.052.

[Haddad 2015] P. Haddad, V. Fischer, F. Bernard, and J. Nicolai, ‘A Physical Approach for Stochastic Modeling of TERO-Based TRNG’, in Cryptographic Hardware and Embedded Systems -- CHES 2015, T. Güneysu and H. Handschuh, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 357–372. doi: 10.1007/978-3-662-48324-4_18.

[Keshner 1982] M. S. Keshner, ‘1/f noise’, Proceedings of the IEEE, vol. 70, no. 3, pp. 212–218, 1982.

[Sunar 2007] B. Sunar, W. Martin, and D. Stinson, ‘A Provably Secure True Random Number Generator with Built-In Tolerance to Active Attacks’, IEEE Trans. Comput., vol. 56, no. 1, pp. 109–119, Jan. 2007, doi: 10.1109/TC.2007.250627.

[Timmer 1995] J. Timmer and M. Koenig, ‘On generating power law noise.’, Astronomy and Astrophysics, vol. 300, p. 707, 1995.

colorednoise F. Patzelt, Colorednoise. 2022. [Online]. Available: https://github.com/felixpatzelt/colorednoise