Security Evaluation of Physical Random Number Generators

Werner Schindler

Federal Office for Information Security (BSI), Bonn

Trégastel, June 3, 2008

Outline

- Introduction
- Design and evaluation criteria for physical RNGs
  - Guess work and entropy
  - Stochastic model
  - Examples
  - Online tests, tot test, self test
- AIS 31 and ISO 18031

General requirements on RNGs (I)

Applications: IVs for block ciphers, challenges, …

Task: The random numbers should prevent (at least) replay attacks and correlation attacks.

Requirement R1:

The random numbers should not show any statistical weaknesses.

Note: R1 is usually verified with statistical tests.

Is Requirement R1 sufficient for sensitive applications?

Sensitive applications

Examples: session keys, signature parameters, ephemeral keys, …

- These random numbers must be kept secret!
- A potential attacker may know preceding or successive random numbers (by challenges, openly transmitted IVs, session keys of messages which he / she has received legitimately, …)

General requirements on RNGs (II)

Requirement R2: The knowledge of subsequences of random numbers shall not allow to *practically* compute predecessors or successors or to guess them with non-negligibly larger probability than without knowledge of these subsequences.

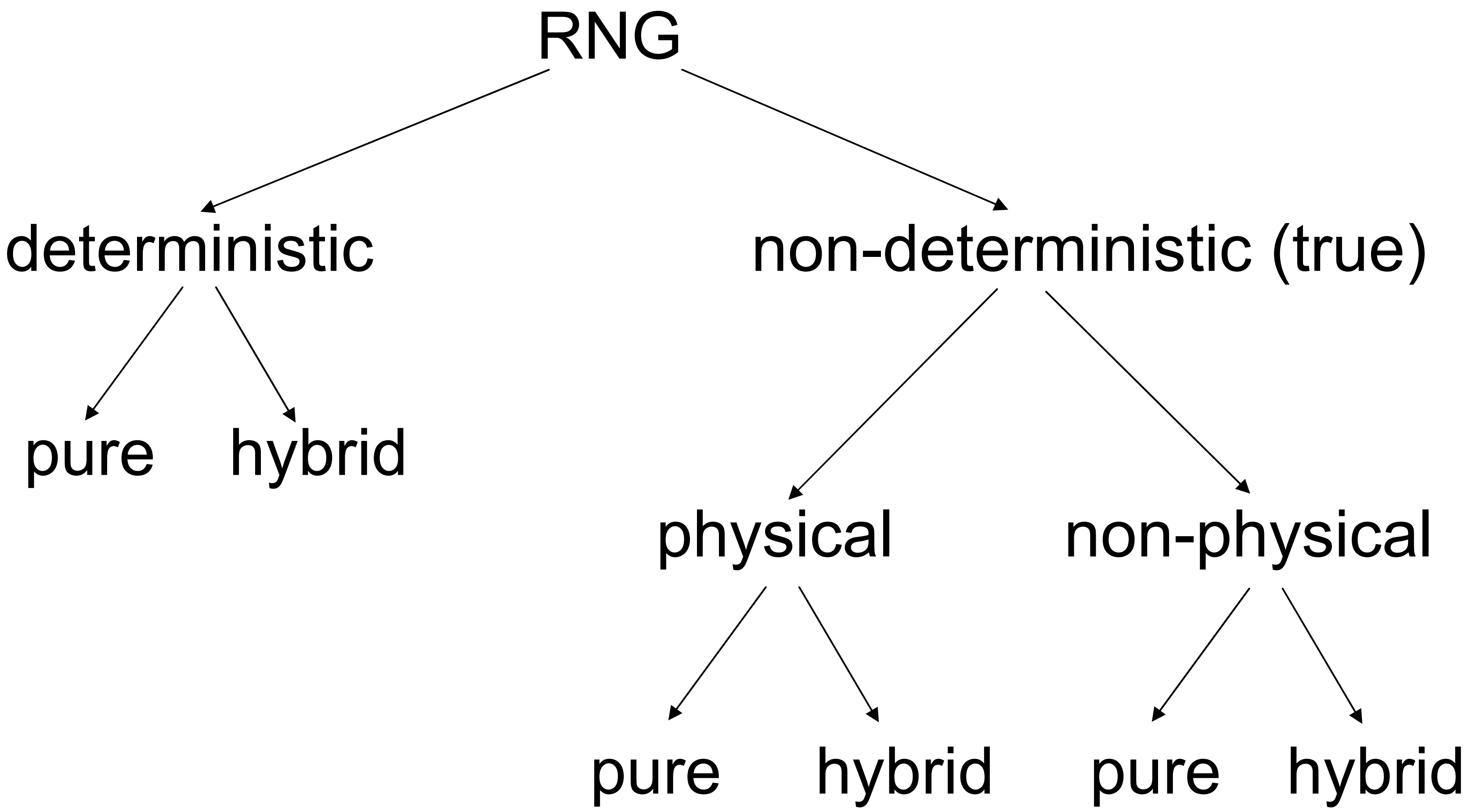
- Requirement R2 is indispensable for sensitive applications

Note: For deterministic RNGs additional requirements exist, which are relevant for particular applications.

Ideal RNGs

- ❒ Even with maximum knowhow, most powerful technical equipment and unlimited computational power an attacker has no better strategy than "blind" guessing (brute force attack).
- ❒ Guessing n random bits costs 2^{n-1} trials in average.
- ❒ The guess work remains invariant in the course of the time.
- ❒ An ideal RNG is a mathematical construct.

Classification of ‘real-world’ RNGs



RNG

deterministic

pure

hybrid

non-deterministic (true)

physical

pure

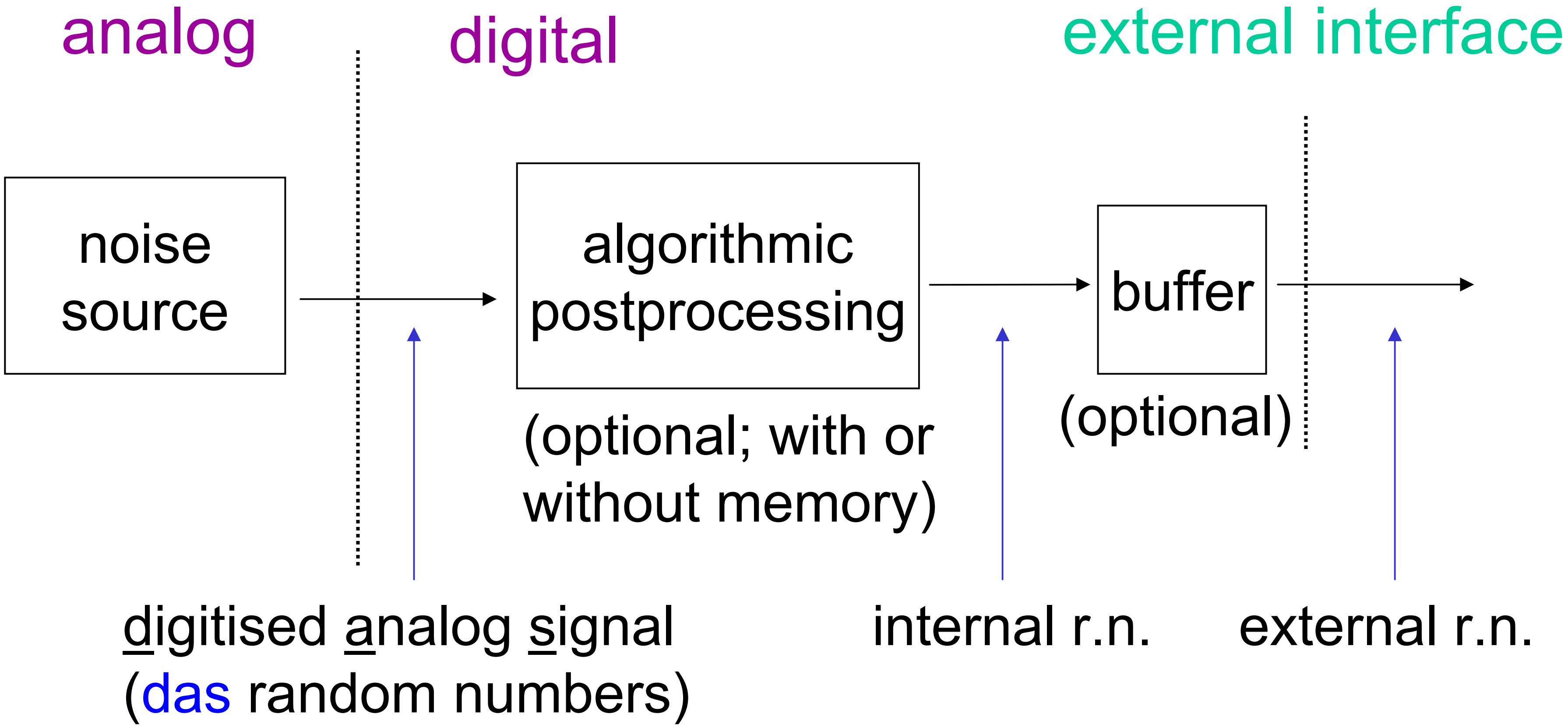
hybrid

non-physical

pure

hybrid

Physical RNG (PTRNG) (schematic design)



Noise source

- The noise source is dedicated hardware.
- The noise source exploits, for example,
  - noisy diodes
  - free-running oscillators
  - radioactive decay
  - quantum photon effects
  - ...

Guess work and entropy (I)

- Let X be a random variable that assumes values in a finite set S = {s_1, ... ,s_t}.
- The optimal guessing strategy begins with those values that are assumed with the largest probability.
- Task of any security evaluation of an PTRNG: Estimation of the guess work
- Note:
  - Deterministic RNGs: The strength may decrease in the course of the time when attacks on the applied cryptographic primitives become feasible.
  - PTRNGs: The workload to find random numbers remains invariant in the course of time.

Entropy (Shannon entropy)

Definition: Let X denote a random variable that assumes values in a finite set S = {s_1, ... ,s_t}. The (Shannon) entropy of X is given by

$$\boxed{H(X) = -\sum_{j=1}^{t} \text{Prob}(X = s_j) * \log_2 (\text{Prob}(X=s_j))}$$

Remark: $0 \leq H(X) \leq \log_2 | S |$

Rényi entropy

For $0 \leq \alpha \leq \infty$ the term

$$H_\alpha(X) = \frac{1}{1-\alpha} \log_2 \sum_{j=1}^{t} \mathrm{Prob}(X = s_j)^\alpha$$

denotes the Rényi entropy of X to parameter α.

For fixed X the Rényi entropy is monotonously decreasing in α.
The most important parameters are $\alpha = 1$ (Shannon entropy) and $\alpha = \infty$ (or more precisely, $\alpha \to \infty$; min-entropy).

Guess work and entropy (II)

- The min entropy is the most conservative entropy measure. For any distribution of X a lower bound for the guesswork can be expressed in terms of its min entropy while the Shannon entropy may suggest larger guess work.

Guess work and entropy (III)

- If $X_1, X_2, \ldots$ denotes a sequence of binary-valued iid (identically and identically distributed) random variables then $H(X_1, X_2, \ldots, X_n)/n \approx \log_2$ (average number of guesses per bit) unless n is too small.
- The assumption "iid" may be relaxed, e.g. to "stationary with finite memory".
- If the random variables $X_1, X_2, \ldots, X_n$ are 'close' to the uniform distribution all parameters α give similar Renyi entropy values.

Guess work and entropy (IV)

❐ (At least) the internal random numbers usually fulfil at least the second and the third condition.

❐ Hence we consider the Shannon entropy in the following since it is easier to handle than the min entropy (→ conditional entropy).

Security evaluation of a physical RNG

Main Steps:

Investigate

- ❒ the RNG design and its implementation
- ❒ the mechanisms to detect eventual failures that cause non-tolerable weaknesses of the random numbers while the PTRNG is in operation

Evaluation of the generic design

- Goal: Estimate the entropy per internal random bit
- Note: Entropy is a property of random variables and not of values that are assumed by these random variables (here: random numbers).
- In particular, entropy cannot be measured as temperature, voltage etc.
- General entropy estimators for random numbers do not exist.

Warning Warning Warning

- The test values of Maurer‘s „universal entropy test“ and of Coron‘s refinement are closely related to the entropy per random bit *if the respective random variables fulfil several conditions*.
- If these conditions are not fulfilled (e.g. for pure deterministic RNGs!) the test value need not be related to the entropy.
- The adjective “universal” has caused a lot of confusion in the past.

- We interpret random numbers as realizations of random variables.
- Although entropy is a property of random variables we will loosely say
  “(average) entropy per random number” instead of “(average) gain of entropy per corresponding random variable”.
- A reliable security evaluation of a PTRNG should be grounded on a *stochastic model*.

Stochastic model (I)

- Goal: Estimate the increase of entropy per internal random number
- Ideally, a stochastic model specifies a family of probability distributions that contains the true distribution of the internal random numbers.
- At least, the stochastic model should specify a family of distributions that contain the distribution
  - of the das random numbers or
  - of ‚auxiliary' random variables

  if this allows to estimate the increase of entropy per internal random number.

Example 1: Coin tossing (I)

- **PTRNG:** A single coin is tossed repeatedly.
  "Head" (H) is interpreted as 1, "tail" (T) as 0.
- **Stochastic model:**
  - The observed sequence of random numbers (here: heads and tails) are interpreted as values that are assumed by random variables $X_1, X_2, \ldots$.
  - The random variables $X_1, X_2, \ldots$ *are assumed to be independent and identically distributed.*
    (Justification: Coins have no memory.)
  - $p := \text{Prob}(X_j = H) \in [0,1]$ with unknown parameter p

Example 1: Coin tossing (II)

Note: A *physical model* of this experiment considered the impact of the mass distribution of the coin on the trajectories.

The specification and verification of a physical model is much more difficult than the specification and verification of the stochastic model.

Stochastic model (II)

- A stochastic model is not a physical model. In particular, it does not provide the exact distribution of the das random numbers or the internal numbers in dependency of the characteristics of the components of the noise source.
- Instead, the stochastic model only specifies a class of probability distributions which shall contain the true distribution (cf. Example 1).
- The class of probability distributions usually depends on one or several parameters.

Stochastic model (III)

- The stochastic model should be verified by theoretical considerations and experiments.
- The parameter(s) of the true distribution is / are guessed on basis of measurements.
- An appropriate stochastic model allows the design of effective online tests that are tailored to the specific RNG design.

Example 1: Coin tossing (III)

Entropy estimation (based on the stochastic model):

- Observe a sample $x_1,x_2, \ldots, x_N$
  Set $\tilde{p} := \#\{j \leq N \mid x_j = H\} / N$
- To obtain an estimate $\tilde{H}(X_1)$ for $H(X_1)$
  substitute p into the entropy formula:
  $\tilde{H}(X_1) = - (\tilde{p} * \log_2 (\tilde{p}) + (1-\tilde{p}) * \log_2(1-\tilde{p}))$

Stochastic model (IV)

- For PTRNGs the justification of the stochastic model is usually more difficult and requires more sophisticated arguments.
- To estimate entropy the parameter(s) are estimated first, and therefrom an entropy estimate is computed (cf. Example 1).
- If the random numbers are not independent the *conditional entropy* per random bit has to be considered.

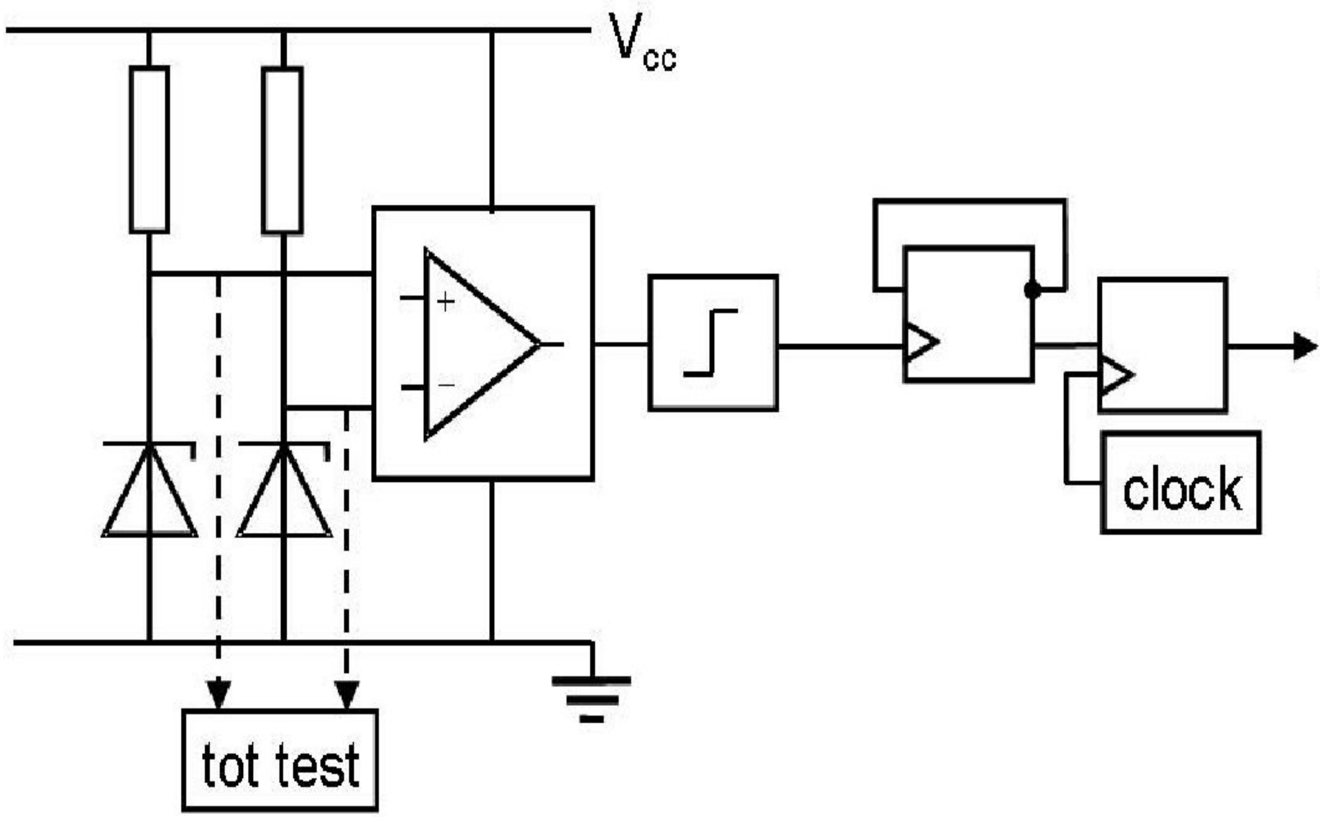
Conditional entropy

Let $X_1,X_2,...$ denote random variables that assume values in a finite set $S = \{s_1, ... ,s_t\}$. The *conditional entropy*

$$\boxed{H(X_{n+1} \mid X_1,...,X_n) = \sum_{x_1,...,x_n \in S} H(X_{n+1} \mid X_1=x_1,...,X_n=x_n) * \text{Prob}(X_1 = x_1,\ldots,X_n = x_n)}$$

quantifies the increase of the overall entropy when augmenting X_{n+1} to the sequence $X_1,...,X_n$.

Example 2



Random number generation (I)

- The Schmitt trigger latches a flip-flop by each 0-1-crossing (up-crossing) of its input voltage

- The clock latches the second flip-flop with constant cycle length, i.e. at time 0, s, 2s,…

$\rightarrow$ internal random numbers: y_1, y_2, ...

Reference: [1] W. Killmann, W. Schindler: A Particular Design for Physical RNGs with Robust Entropy Estimators. To appear in the proceedings of CHES 2008.

Random number generation (II)

- das random number r_n := number of 0-1-crossing in time period ((n-1)s,ns]
- internal random number y_n = $y_{n-1}+r_n = y_0+r_1 + ...+r_n$ (mod 2)
- **Goal:** Determine a lower bound for the conditional entropy $H(Y_{n+1} \mid Y_1, ...,Y_n)$
- First step: Study the stochastic process R_1, R_2,...

Stochastic model (I)

❒ We interpret the intervals $t_1,t_2,\ldots.$ between subsequent 0-1-crossings of the comparator input voltage as realizations of random variables $T_1,T_2,\ldots$

❒ Shortly after start-up of the RNG the noise source should be in equilibrium state.

❒ → The stochastic process $T_1,T_2,\ldots$ is assumed to be stationary (but not necessarily independent).

Stochastic Model (II)

The corresponding random variables meet the following conditions / relations:

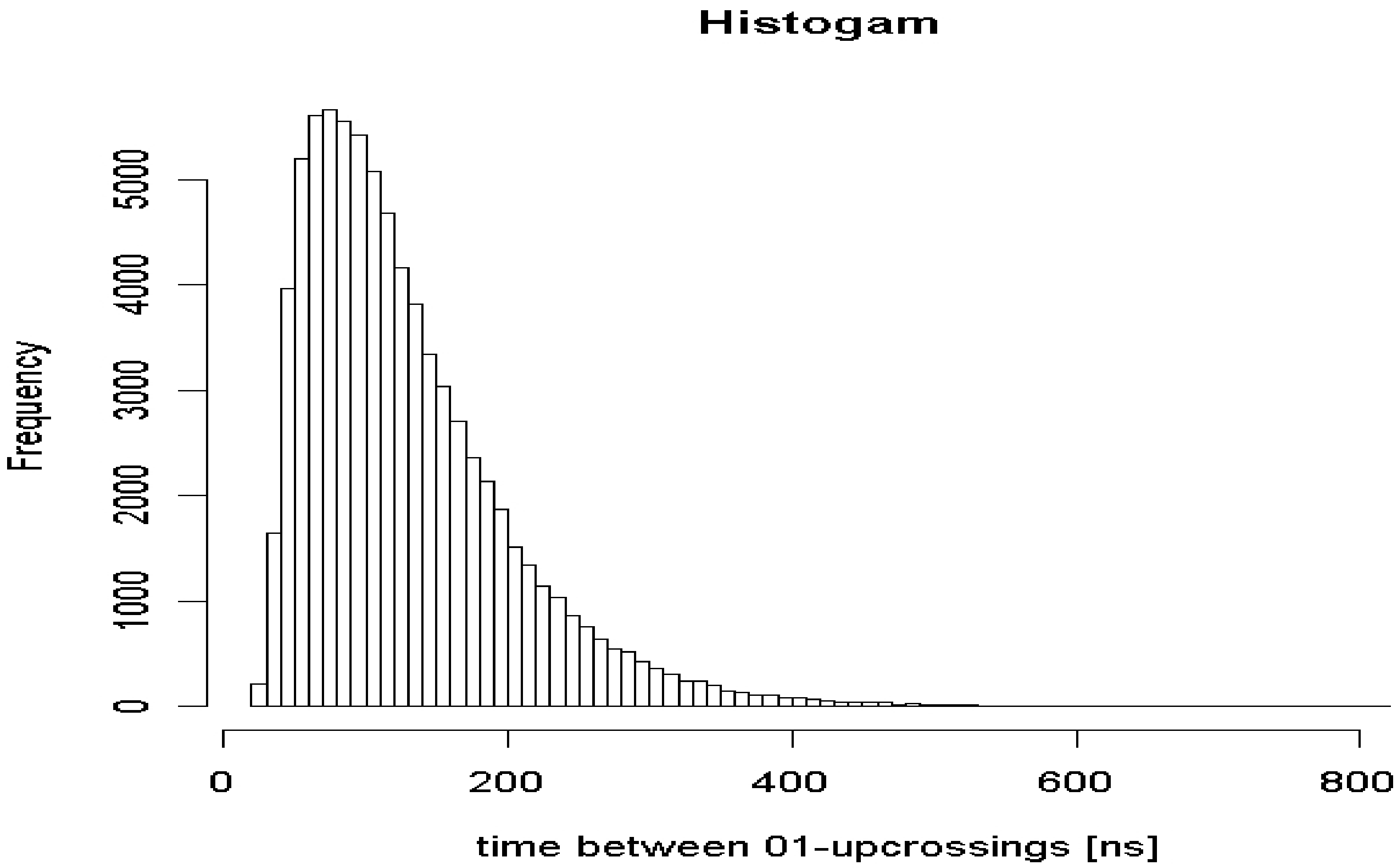
a)$T_1, T_2, \ldots$ are stationary

b) $R_n := Z_n - Z_{n-1}$ for

c) $Z_n := \min \{m \in N \mid W_0 + T_1 + \ldots + T_m > sn\}$

with $W_n := Z_n - ns$

RNG prototype: Empirical distribution of T_j



Remarks

❐ Interestingly, a) to c) fits to several RNG designs. This makes it profitable to study this scenario.

❐ Even if the das random numbers of different RNG designs can be modeled in this way the distribution of the random variables T_1,T_2,…and thus of R_1,R_2,.. and Y_1,Y_2,… may be very different.

„Transfer“ of stationarity

It can be shown that (under mild assumptions) the stationarity of the random variables $T_1,T_2,\ldots$ implies the stationarity of the random variables

$W_0,W_1,W_2,\ldots$,

$R_1,R_2,\ldots$ (das random numbers) and

$Y_1,Y_2,\ldots$ (internal random numbers)

Auxiliary random variables

Definition

- ❒ $V_{(u)} := \inf \{d: T_1 + T_2 + \ldots T_{d+1} > u\}$
- ❒ $\mu = E(T_j)$
- ❒ σ^2 := generalized variance of $T_1, T_2, \ldots$
- ❒ $\Phi(\,.\,)$:= cumulative distribution function of the standard normal distribution

Auxiliary random variables (II)

Lemma: For u = v · μ the Central Limit Theorem yields the approximations

$$\Pr ob(V_{(v\mu)} = k) \approx \Phi\left(\frac{v-k}{\sqrt{k}} \cdot \frac{\mu}{\sigma}\right) - \Phi\left(\frac{v-(k+1)}{\sqrt{k+1}} \cdot \frac{\mu}{\sigma}\right) \quad for \quad k \geq 1$$

$$\Pr ob(V_{(v\mu)} = 0) \approx 1 - \Phi\left((v-1)\frac{\mu}{\sigma}\right)$$

Main results (I)

Theorem 1: (i) Let G_W denote the stationary distribution of the random variables $W_0, W_1, W_2, \ldots$. Then

$$E\left(\left(R_1 + R_2 + \cdots R_j\right)^k\right) \approx \int_0^{js} E\left(\left(V_{(js-u)} + 1\right)^k\right) G_W(du)$$

with equality if the sequence $T_1, T_2, \ldots$ is independent.

In particular, this formula can be used to compute the autocovariance function of $R_1, R_2, \ldots$

Main results (II)

Theorem 1 (ctd’; special case) If the random variables T_1, T_2, ... are independent then

$$H\left(Y_{n+1} \mid Y_0, \ldots, Y_n\right) \geq \int_0^s H\left(V_{(s-u)}\right)(\mathrm{mod}\,2)\; G_W\left(du\right)$$

If, additionally, T_j has a continuous cumulative distribution function F(.) then G_W has density $g_W(w) := (1-F(w)) / \mu$.

(see [1] for further related results)

RNG prototype: Experimental results

- Entropy per internal random bit > $1-10^{-4}$.
- Output rate of $5*10^5$ random bits / sec is principally possible.

PTRNGs in operation: Potential risks

❒ Worst case: total breakdown of the noise source

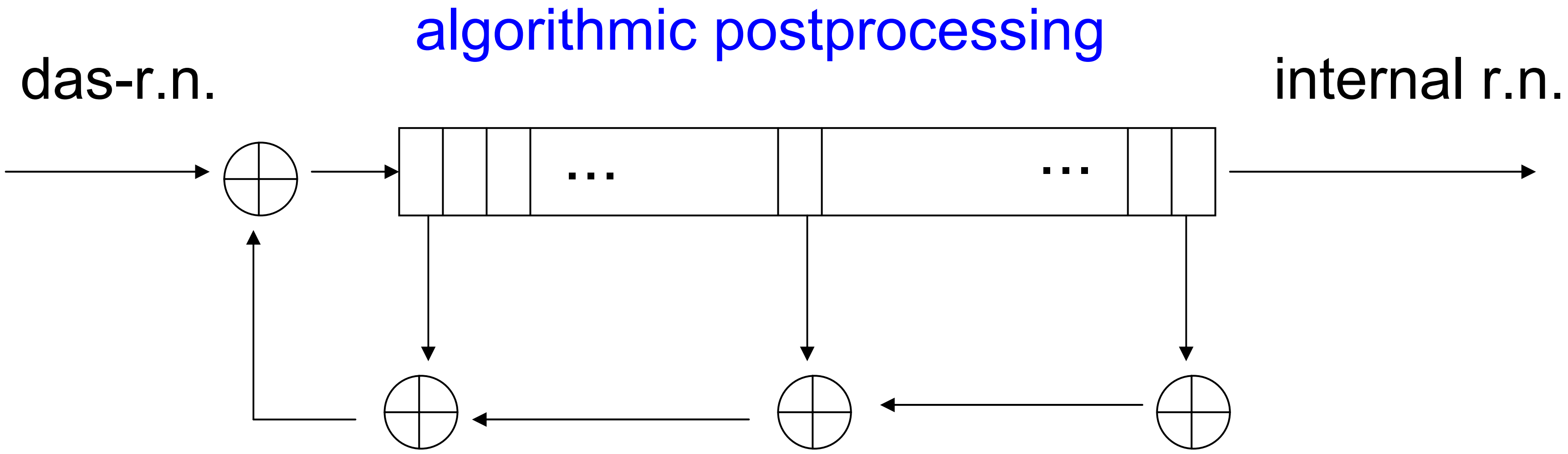
❒ Ageing effects, tolerances of components of the noise source and external influences might cause the generation of random numbers with unacceptably low quality.

Such events must be detected with certainty so that appropriate responses can be initiated.

Security measures

	goal
tot-test	shall detect a total breakdown of the noise source very soon
startup test	shall ensure the functionality of the PTRNG when it is started
online test	shall detect non-tolerable weaknesses of the random numbers sufficiently soon

Example 3: LFSR



worst case scenario: total breakdown of the noise source, inducing constant das-random numbers

→ entropy / das bit = 0, ... **but** ...

internal r.n.s: good statistical properties!!!

Example 3 (II)

- Statistical blackbox tests that are applied to the internal random numbers will not even detect a total breakdown of the noise source (unless the linear complexity profile is tested).
- Instead, the online test should be applied to the das random numbers (typical situation).

Online tests: General remarks

❐ The online test should be tailored to the particular RNG (→ stochastic model).

❐ For Example 1 (coin tossing) a monobit test were appropriate.

❐ Since the online test(s) is (are) usually realized by statistical test(s), also „false” noise alarms may occur.

❐A failure of the online test causes a noise alarm

Security evaluation

- ❐ A reliable security evaluation shall verify the suitability of the online test, the tot test and the startup test.
- ❐ The evaluation also comprises the specified consequences of a noise alarm (e.g.: RNG is shut down, audit of the noise alarms, restart of the RNG by a human operator, …).

Common Criteria (CC)

- ❐ provide evaluation criteria for IT products which shall permit the comparability between independent security evaluations.
- ❐ A product or system that has successfully been evaluated is awarded with an internationally recognized IT security certificate.

AIS 31 (I)

The Common Criteria and the corresponding evaluation manuals *do not* specify evaluation criteria for random number generators.
In the German evaluation and certification scheme the evaluation guidance document

AIS 31: Functionality Classes and Evaluation Methodology for Physical Random Number Generators

has been effective since September 2001

AIS 31 (II)

- ❐ The AIS 31 is technically neutral. The applicant for a certificate has to give evidence that the PTRNG meets specified requirements.
- ❐ The AIS 31 has been well-tried in many product evaluations.
- ❐ A reference implementation of the applied statistical tests can be found

www.bsi.bund.de/zertifiz/zert/interpr/ais_cc.htm

Alternative security paradigm

- ❐ Crucial points of an AIS 31 evaluation are the understanding of the design and the effectiveness of the online test.

Alternative approach (e.g., ANSI X9.82, Part 2 (draft)):

- ❐ main security anchor: complex algorithmic postprocessing algorithm with memory that meets requirements R1,R2 and R3 (one-way property of the state transition function);
lower requirements on the understanding of the design and the online tests.

Alternative security paradigm: Advantages and disadvantages

- ❐ (+) lower requirements on the understanding of the RNG design
- ❐ (+) lower requirements on the effectiveness of the online tests
- ❐ (-) requires a time-consuming postprocessing algorithm
- ❐ (-) possibly (without being noticed!) only practical security
- ❐ (-) requires the protection of the internal state

ISO / IEC 18031 „Random Bit Generation“

- ❒ covers all classes of RNGs
- ❒ PTRNGs: Allows design principles that either follow the AIS 31 or the ANSI X9.82-2 (draft) approach
- ❒ considers also the correctness of the implementation

Final remark

Combining

- ❒ a strong noise source
- ❒ with effective online tests
- ❒ and a strong algorithmic postprocessing algorithm

provides two security anchors which shall ensure theoretical security and computational security, respectively.

Contact

Federal Office for Information Security (BSI)

Werner Schindler
Godesberger Allee 185-189
53175 Bonn

Tel: +49 (0)3018-9582-5652
Fax: +49 (0)3018-10-9582-5652

Werner.Schindler@bsi.bund.de
www.bsi.bund.de
www.bsi-fuer-buerger.de