Random Numbers in Cryptography

Martin Šimka

Sentivision, Poland / Technical University of Košice, Slovakia

Miloš Drutarovský

Technical University of Košice, Slovakia

Viktor Fischer

Université Jean Monnet, France

About Me

- Former PhD student of Technical University of Košice (Slovakia)
- Short time stages (3-4 months):
  - CoSy group (Prof. Ch.Paar), COSIC group (Prof. B.Preneel and Prof. I.Verbauwhede), LTSI Lab
- Currently working in 

  - IPTV and VOD technology provider

Contents

- Role of randomness in cryptography
- Applications of random values in cryptographic algorithms
- Different requirements set to random numbers in dependency on usage
- Generation of random numbers
- Most known randomness extraction techniques in digital systems
- Testing and certification process

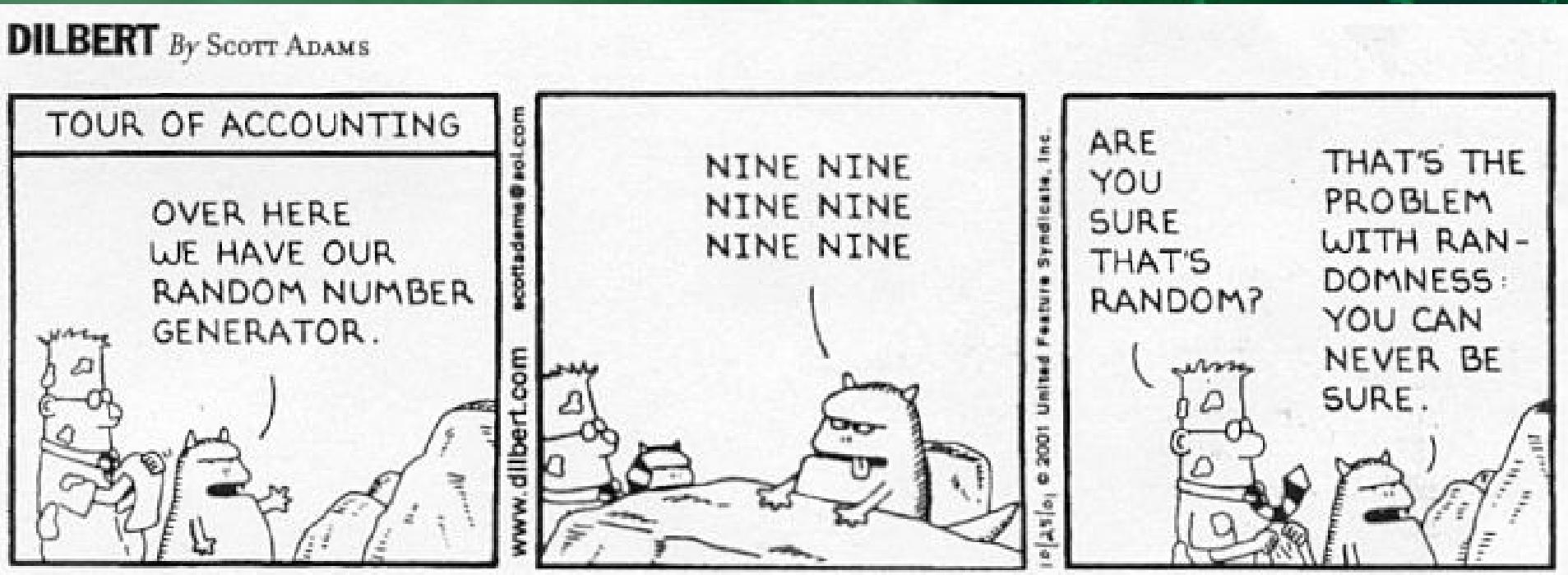
Random Numbers

- **Wide application in several fields:**
  - **Monte Carlo simulation method**
  - **Spectrum spreading telecommunication systems**
  - **Generation of primes**
  - **Cryptography**
  - **Computer games**
  - **Gambling industry**

What Is Random? 1/2

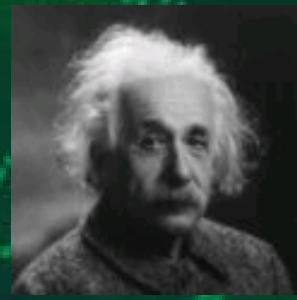
- How to prove randomness?
- Exact definition is missing
- For *finite* numeric sequence – impossible task
  - What is more random?
  - 00110011 or 10011100?

What Is Random? 2 / 2



Is There Any Randomness?

- **Philosophical question**
- **God does *not* play dice with the universe – Albert Einstein**



- **Several physical phenomena present in nature are supposed to be random: quantum mechanics, noise…**

Coin Flipping

- Generation of random sequence – tale 0, head 1
- Truly random or just difficult-to-describe system?
- Randomness appears due to “instability” of the system

Randomness Definitions

- Truly random number is generated by a process, whose outcome is:
  - unpredictable, and
  - which cannot be subsequentially reliably reproduced

Unpredictable sequence

- For unpredictable bit sequence it is computationally infeasible to predict what the next random bit will be, given
  - complete knowledge of the algorithm,
  - the hardware generating the sequence, and
  - all of the previously generated bits

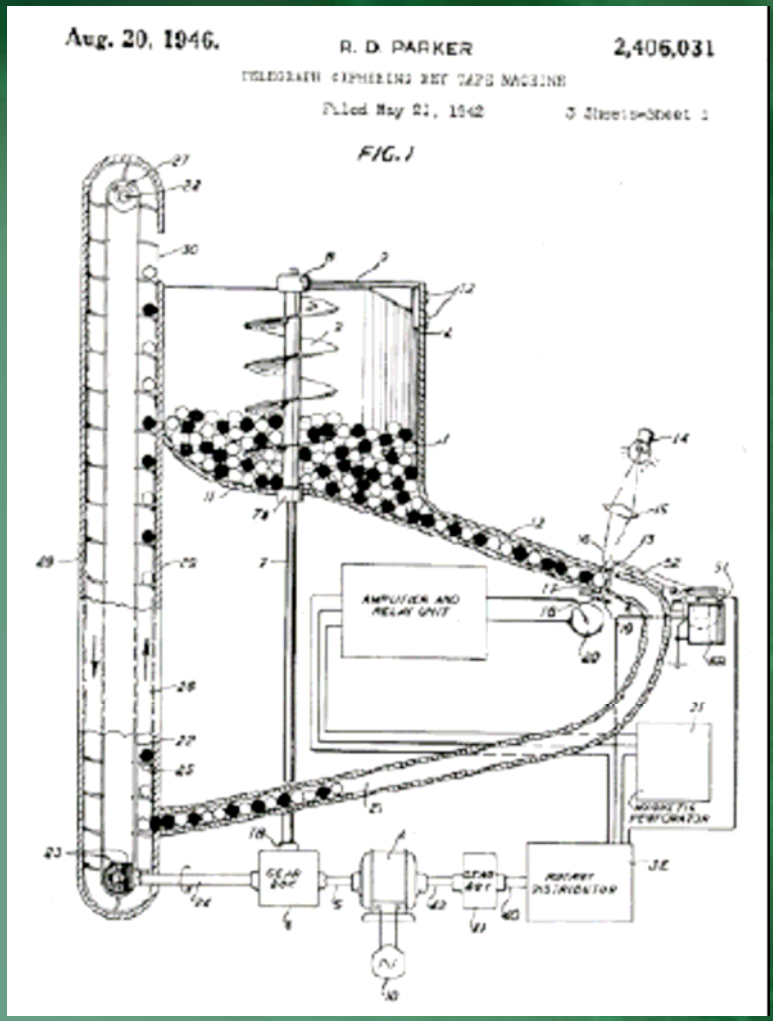
Unrepeatable sequence

- Given the same working conditions (or seed) the generated sequences are completely different

Practical Randomness

- Sequence that has the same statistical properties as random bits, is unpredictable and cannot be reliably reproduced
- Evaluation on how the tested sequence shares the statistical properties of ideal random sequence

History of RNG



- Patented ATT machine from year 1941 generating random sequence
- Source of randomness – large container of black and white balls

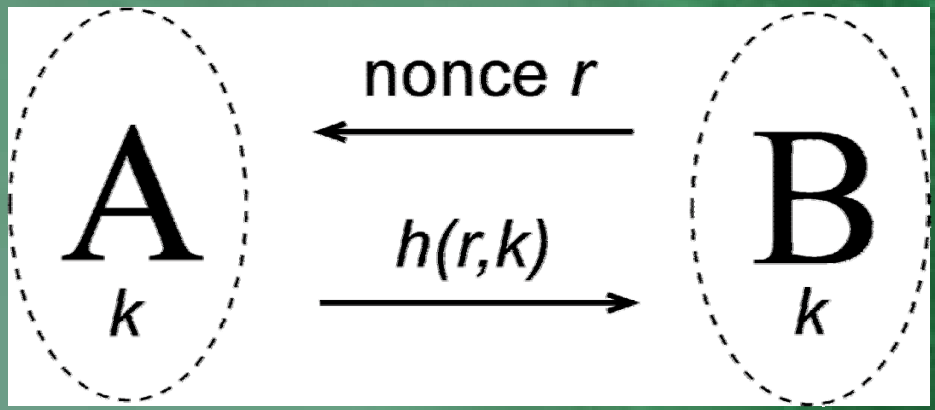
Sources of randomness

- Decay of radioactive material, quantum mechanics processes
- Thermal and other types of noise
- Frequency instability of free-running oscillators

Randomness in Cryptography

- Generation of:
  - Nonce (only once used number)
  - Key
  - Challenge
  - Initialization vector
  - Padding byte
  - Blinding value

Example From Cryptography



- Challenge-response scheme based on keyed one-way function
- Authentication of the entity A to the verifier B
- Random nonce *r*, shared secret *k*, MAC function *h()*

What does assure the randomness?

- For brute-force attack there are no preferred values, all values have equal probability to be keys
- Any bias (difference from ideally random sequence) that would provide additional information on subsequent generated values

Requirements on Random Values

- In dependency on application the random numbers have to be:
  - Unpredictable, un-guessable, non-correlated, statistically equal (have equal probability)
  - Unrepeatable
  - Secret

Types of RNG

- ***TRNGs (True Random Number Generators)*** **– generators of truly random numbers, called also physical generators**
- ***PRNGs (Pseudo-random Number Generators)*** **– generators of pseudo-random sequencies, denoted also as deterministic generators**
- ***Hybrid Generators*** **– combination of above principles**

Random Number Generators

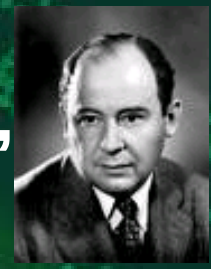
- PRNG – Pseudo-RNG
  - Good statistical properties
  - High output rates
  - Requirement of random seed
- TRNG – True-RNG
  - Natural sources may have statistical defects
  - Post-processing causes lower output rates
  - Complicated implementation in digital systems

Pseudo-random

- Sometimes statistical properties are prioritized – no bias, equiprobability…
- Need to reproduce the sequence in other time, by other part of system
- Predictability could be acceptable
- Looks like random, but is not truly random – pseudo-random

PRNG – deterministic

- Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin (John von Neumann)
- A random seed is expanded to long random-looking sequence
- Security depends on sequence period, internal algorithm
- Knowing internal structure and status or previous values, the output value can be set

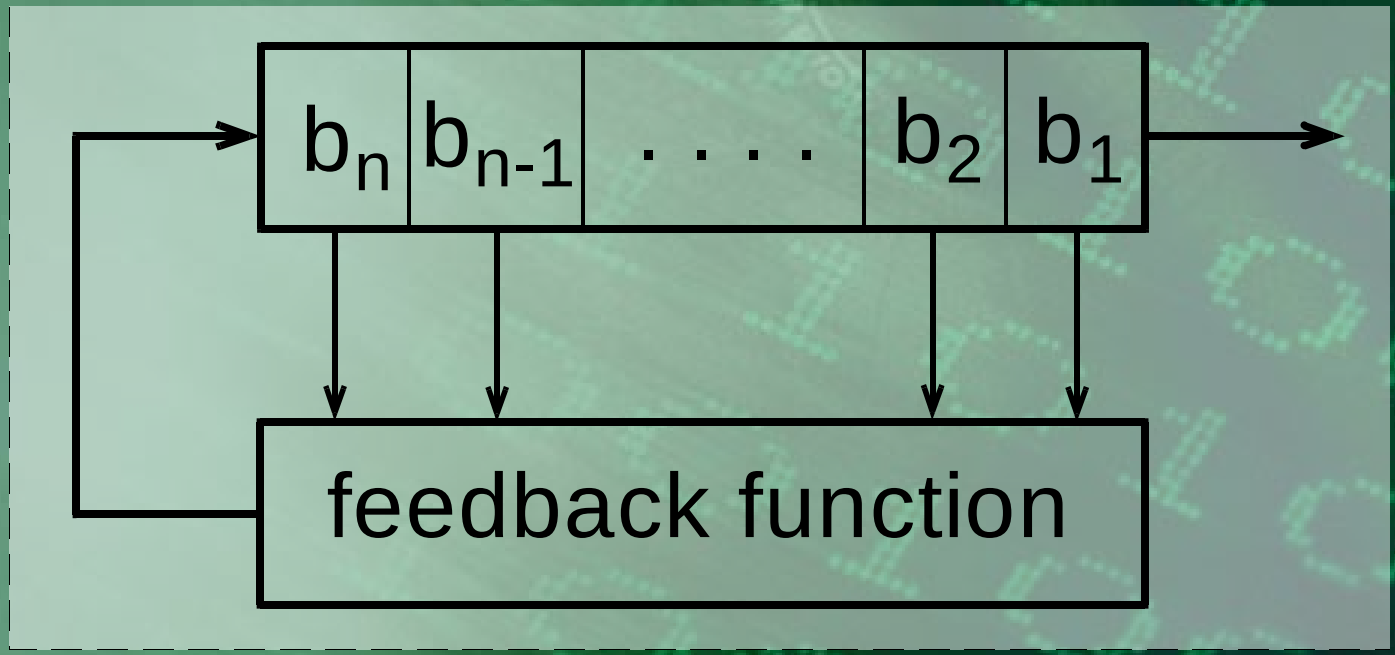


PRNG cont.

- PRNG based on:
  - LFSR
  - Hash functions
  - RSA
  - Cellular Automata
  - Quadratic residua
  - ...

Feedback Shift Register

- Linear for XOR as feedback function
  - Fibonacci and
  - Galois configuration



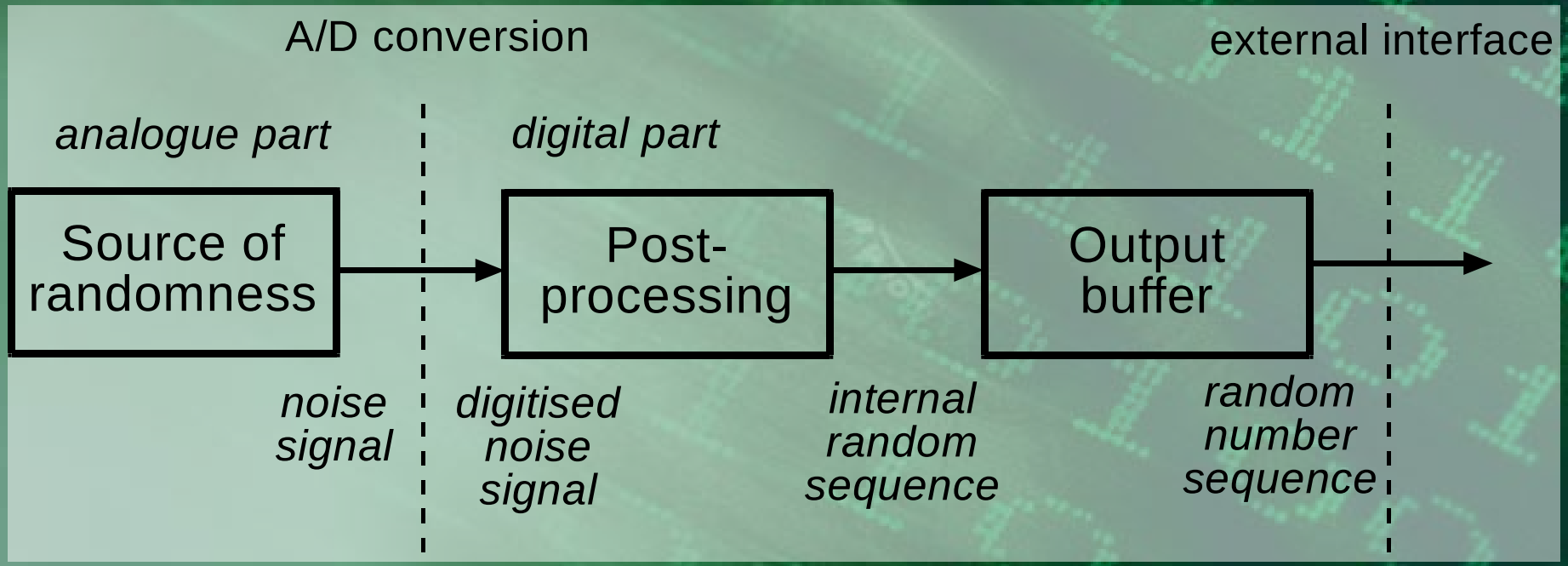
Cryptographically secure pseudo-random

- Knowing the algorithm and the previous output it is computationally infeasible to predict the subsequent output
  - Blum-Blum-Shub
  - Stream ciphers

TRNG

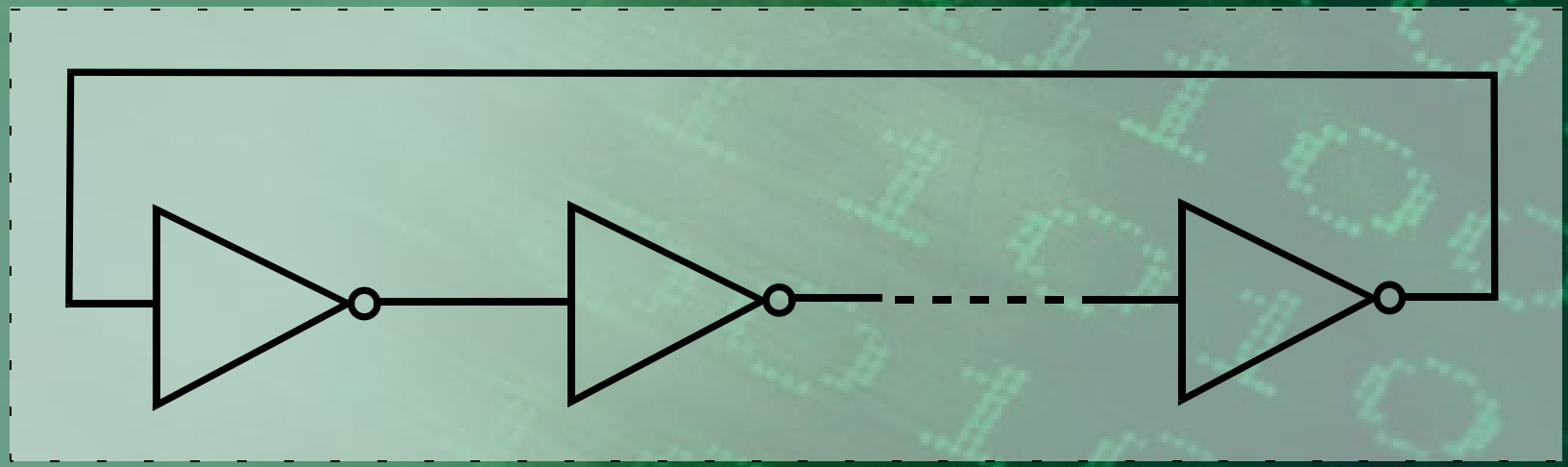
- Based on physical source of randomness – entropy, that is impossible to describe in deterministic way
- Distinguish from quasi random sources in computers: mouse pointer movements, access time of hard disc, system time – these are difficult to describe, low-entropy sources

Typical block diagram of TRNG



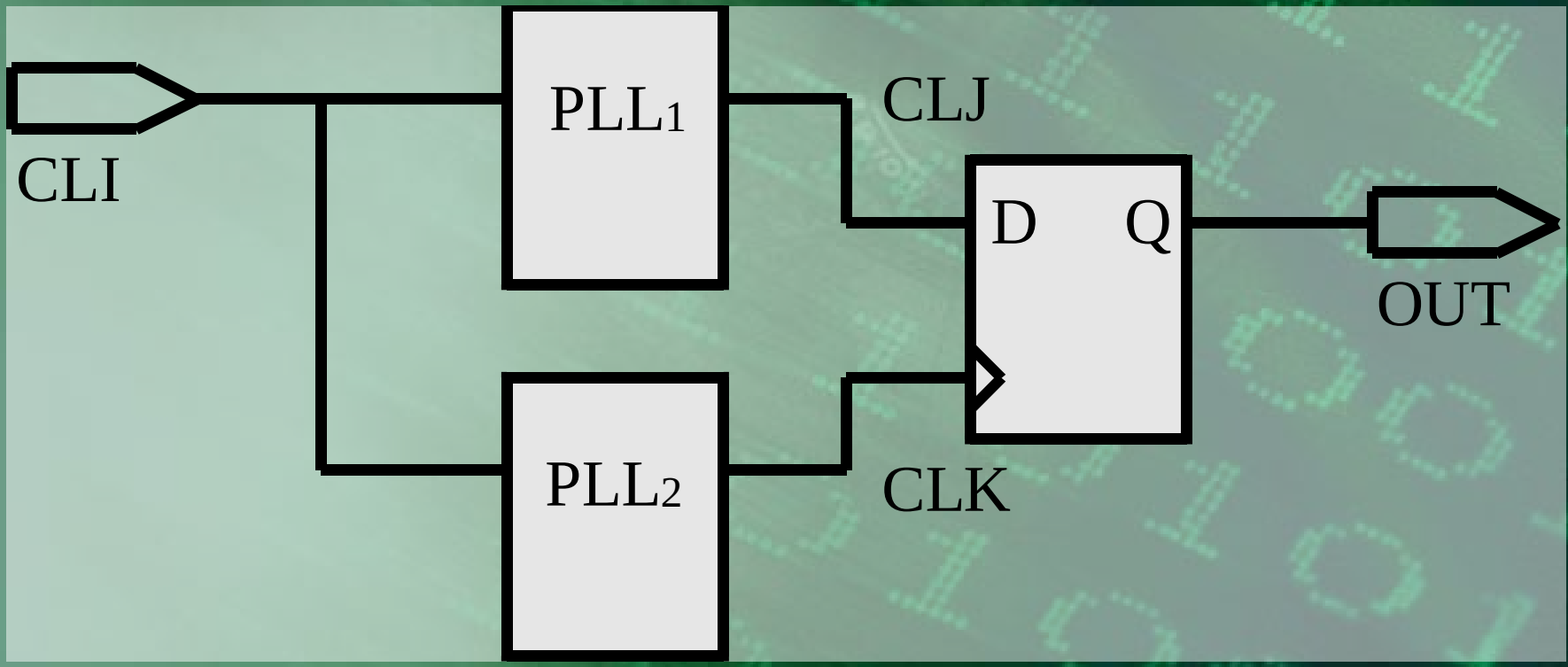
Ring oscillator

- Chain of *odd* number of invertors
- Delayed feedback from the last NOT gate to the first one causes oscillation



PLL-based TRNG

- Stable ratio between CLK and CLJ
- Sampling tracking jitter of PLL



Hybrid RNG

- Slow TRNG seeds PRNG
- Important is to divide clearly both system
  - Mixing truly random values with low entropy and pseudo-random values does not provide acceptable solution
- Offers high bit-rate output based on truly random seed

Internal status of RNG

- PRNG:
  - Previous output value
  - Directly influences the subsequent output value
- Crypto secure PRNG
  - Hidden, not revealed with output sequence
- TRNG
  - No internal status / memory
  - The output depends only on current physical conditions, not on previous output

Entropy

- In PRNG
  - Given by entropy of the seed
  - Constant
- In TRNG
  - Increasing by each generated random value
  - Need to set level of entropy of the source

Entropy cont.

- The entropy does not give a guaranty for randomness
  - Zipped file has high level of entropy (information), however, the archived content is fully deterministic
  - On other hand, if sequence can be compressed, it is not truly random

Implementations of TRNG

- FPGA
  - Xilinx:
    - Kohlbrenner and Gaj (ring oscillators)
    - Schellekens et al. (ring oscillators)
    - Tsoi et al. (external RC circuit)
    - Golic (Fibonacci and Galois ring oscillators)
  - Altera and Actel:
    - Fischer, Drutarovský, Šimka (PLLs)
- Industrial solutions
  - Intel motherboard chipset
  - VIA processors

Netscape case of unsecure RNG

- Wrong implementation of RNG in Netscape for SSL encryption protocol
- Seed value of the PRNG was derived only from 3 parameters:
  - Time of day,
  - ID of running and
  - ID of parent process

Concepts of TRNG

- Sunar et al.
  - Model of entropy collection process
  - Specification of TRNG parameters based on model
- Bucci and Luzzi

Certification process

- **FIPS 140-2 standard (NIST)**
  - **Originally 4 included tests (monobit, poker, runs, long runs on 20.000 bits) were removed**
- **AIS standards (German Federal Office for Information Security (BSI))**
  - **AIS 21 for PRNG**
  - **AIS 31 for TRNG**
  - **Model of randomness extraction**
  - **Analysis of digitalized analogue samples**

Statistical tests

- NIST test suite
- AIS
- DIEHARD tests by G. Marsaglia
- Special tests suited to extraction method

Quastions & Answers


Thank you!

References

- NIST test suite
  - http://csrc.nist.gov/rng/rng2.html
- AIS 31
  - http://www.bsi.bund.de/zertifiz/zert/interpr/ais_cc.htm
- RFC 1750
  - http://www.ietf.org/rfc/rfc1750.txt