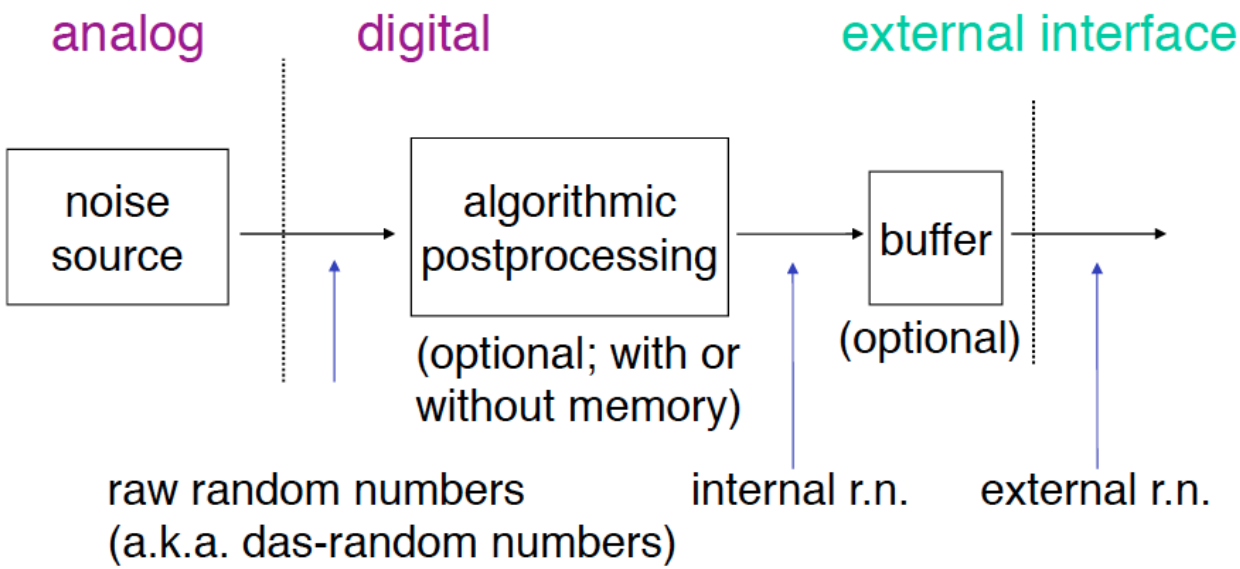
True Random Number Generators



True RNGs

- A noise source is a special type of entropy source consisting of dedicated hardware
- Typical noise sources:
  - Diodes
  - Free-running oscillators
  - Ring oscillators
  - Quantum photon effects

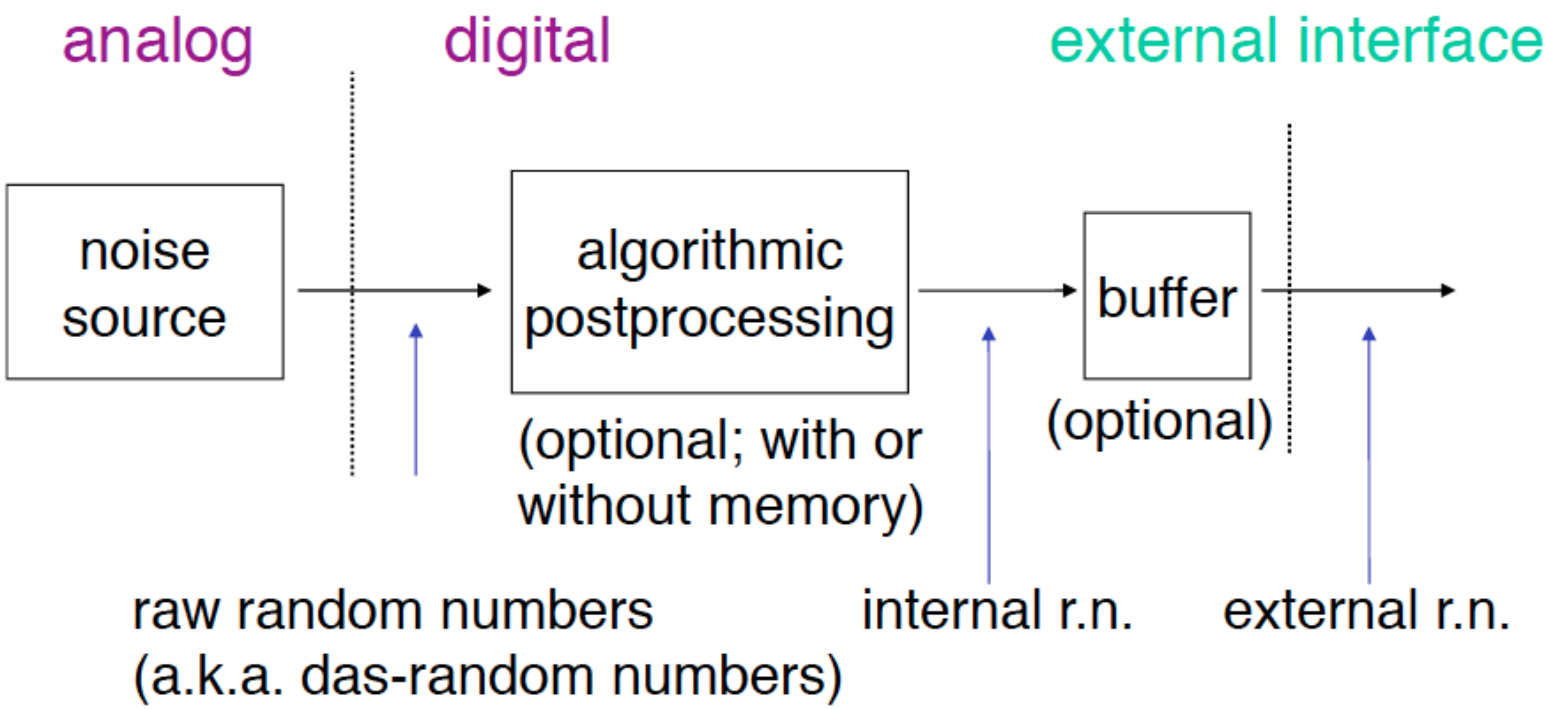
True RNGs

- True RNGs exploit the randomness in some physical phenomena
- Typical examples:
  - Elapsed time between emission of particles during radioactive decay
  - Thermal noise from a semiconductor diode or resistor
  - The frequency instability of a free running oscillator
  - The amount a metal-insulator semiconductor capacity is charged during a fixed period of time
  - Air turbulence within a sealed disk drive which causes random fluctuations in disk drive sector read latency times
  - Sound from a microphone or video from a camera

Software-based RNGs

- Designing a random bit generator in software is even more difficult than doing so in hardware
- These processes have underlying hardware components, but these are under control
- Such are sometimes called **non-physical TRNGs**
- Processes upon which software random bit generators may be based include:
  - the system clock
  - elapsed time between keystrokes or mouse movement
  - content of input/output buffers
  - user input
  - operating system values such as system load and network statistics

True RNGs


analog
digital
external interface
noise source
algorithmic postprocessing
buffer
(optional; with or without memory)
(optional)
raw random numbers
(a.k.a. das-random numbers)
internal r.n.
external r.n.

Requirements R1-R4

- Requirement R1 can be checked using statistical (diehard) tests
- If internal random numbers are unpredictable, TRNG meets R2
- Except for some unusual designs R3 and R4 are automatically fulfilled

DRNGs vs TRNGs

- A pure DRNG fulfills Requirement R2 and possibly R3 if the state function and output function are sufficiently complex
- A pure DRNG provides practical (computational) security
- Its security assessment may change in the course of time

- A TRNG fulfills R2 if the entropy of the internal random numbers (the distribution of the underlying random variables) is sufficiently large
- This implies theoretical (time-invariant) security
- Secure forever :)

DRNGs vs TRNGs

- In practice: coming up with high quality and well controlled noise sources is challenging
- Several proposals have failed over the years, due to entropy properties or due to the use of simple structures for algorithmic post-processing
- We are stuck between a rock and a hard place:
  - DRNG: Security properties may change over time
  - TRNG: Low-cost, high-quality, and high-bandwidth entropy sources that can be integrated on to our chips are scarce
- Hybrid DRNGs: low-cost and low-bandwidth entropy sources, coupled with cryptographically strong and high-bandwidth functions for DRNGs, whose design parameters can be selected and controlled