Physical Entropy Sources

Entropy Source Validation Workshop

April 29, 2021

Chris Celi (NIST)

John Kelsey (NIST)

Outline

- “Designed” and “found” entropy sources
- Design Considerations
- Modeling
- Failure Conditions

Physical Entropy Sources

- A more encompassing term might be a “designed” entropy source
  - Not just taking what is at the device’s disposal
  - Intentionally designing and including components to capture entropy
  - Ring oscillators, captured light sources…
- In contrast with a “found” entropy source
  - Using metrics and elements that are already part of the system
  - Timers, devices, interrupts…

Physical Entropy Sources

- Why the distinction?
- A “found” entropy source such as a camera, microphone or accelerometer might…
  - Not always be functioning or available
  - Become outdated based on available technology
    - Hard drive access timings from spinning disk drives
  - Be susceptible to other operations on the device
    - Primary use of the entropy source is not to generate entropy

Physical Entropy Sources

- Why the distinction?
- A “designed” entropy source…
  - Can have a probability model in mind from the designer
  - Can be more directly tested and observed

Physical Entropy Sources

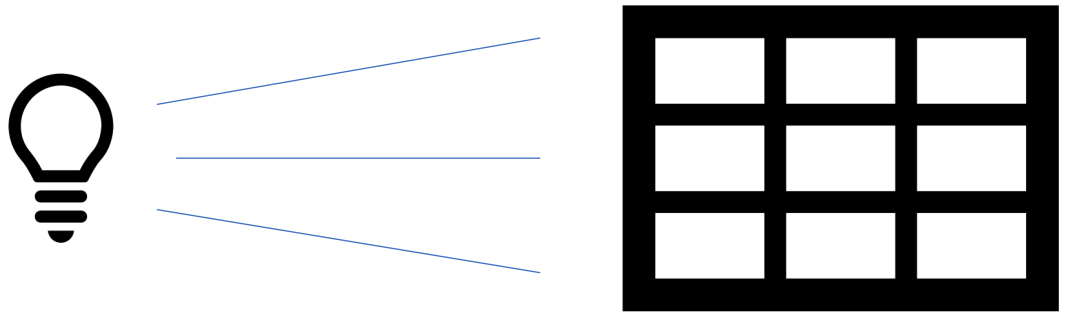
- Why the distinction?
- A “designed” entropy source is often preferred and easier to work with
- Depends on some physically nondeterministic process
- Can be selected because it provides a simpler model, and is easy to test

What Is Being Measured?

- A camera in a closed lightless box is exploiting thermal noise affecting the image capture
- A ring-oscillator also captures thermal noise causing skew with the independent clock cycles

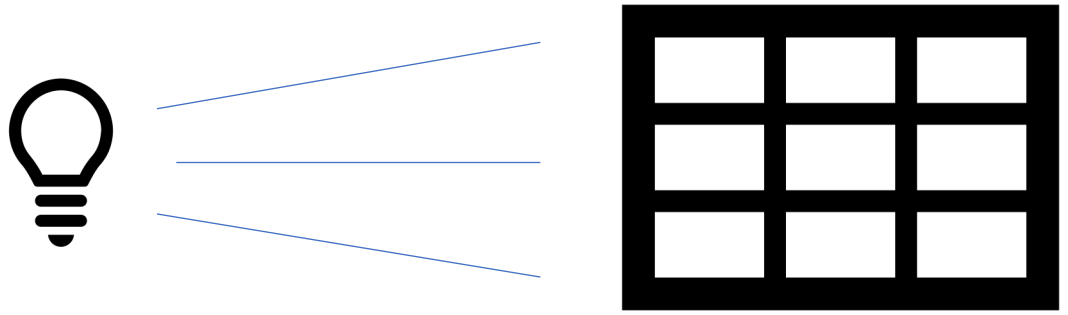
- Understanding these sources can help determine a model

Quantum Source Example



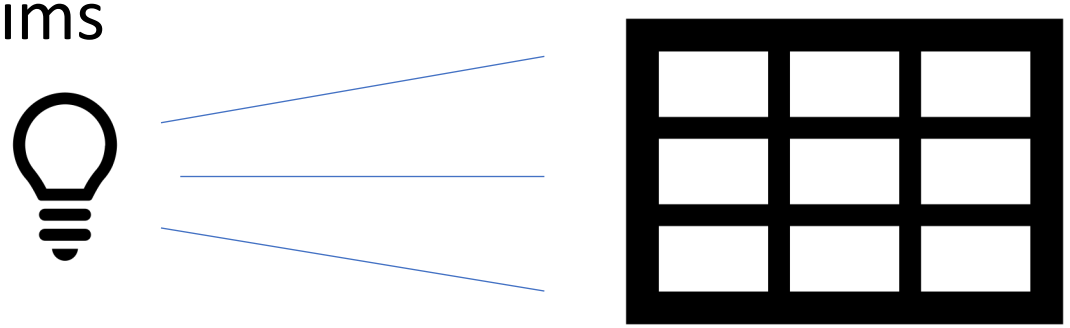
Quantum Source Example

- Multiple ways to capture quantum data
- Where on the grid a photon is detected?
- How many photons in a time period?
- How long between photons?



Quantum Source Example

- How to model the source?
- Number of photons arriving in a time window follows a Poisson distribution
- Literature can be cited for these claims
- With a model in mind, certain parameters can be selected based on the design to lead to a min-entropy estimate



Design Considerations

- Design with these processes in mind from the beginning!

- Data collection
- Practicality
- Health tests and failure conditions

Data Collection

- **MUST** be able to grab raw noise bits
  - Can be through a debug flag or debug mode
  - Justify how capturing the bits does not impact the entropy source behavior if additional options are used to obtain the bits
- **MUST** be able to grab bits from each non-vetted conditioning component
- Be aware of the restart test requirements
  - 1000 bits each from 1000 restarts
- “Normal operating conditions”
- SP800-90B Section 3.2.4
- FIPS 140-2 IG 7.18

Practicality

- Models become more complicated when elements are added
- Think about how the model is impacted while adding design elements to counteract common pitfalls
- If multiple copies of the same entropy source are used, how are they combined?
  - Concatenation could lead to periodicity concerns
  - XOR could cancel out high entropy bits

Health Tests

- RCT and APT must occur on all outputs
- Developer-defined health tests can also be performed
- How will the device process when it detects failures?
  - Intermittent failures?
  - Persistent failures?
- Where do the health tests occur when multiple copies of the same entropy source are used?
  - As a part of each copy of the entropy source?
  - On the full output?
  - In the case of ring oscillators, each would need to be tested to avoid worst-case scenarios like a locked state

Modeling

- Find what type of entropy is being measured by the device
  - Thermal? Quantum?
- Find a probability distribution
  - Easier to do with a simple model
  - H[min] = -log(P[max]) does not fully describe the distribution
  - Sometimes certain properties need to be measured
  - May come from cited sources
- Heuristically, it may be helpful to consider an attacker
  - Give the attacker more information than they would ever have naturally, and prove an advantaged attacker cannot guess the next output reliably

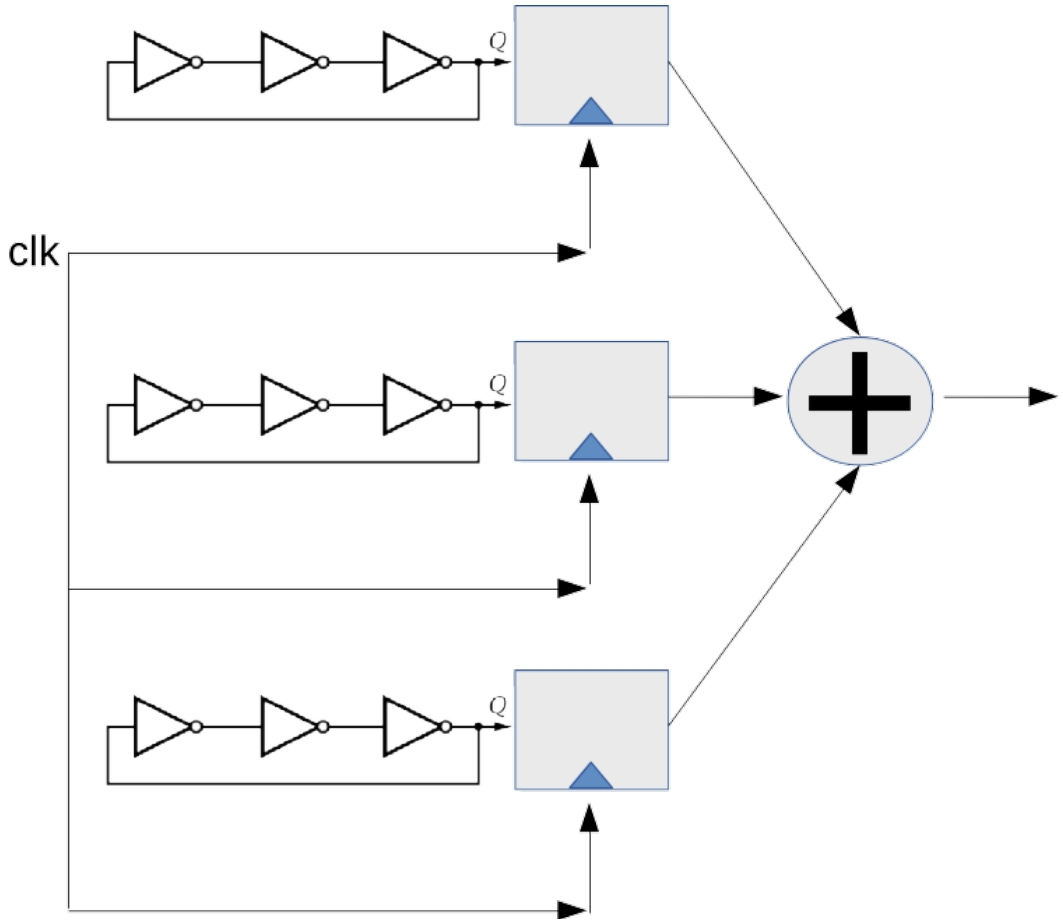
Failure Conditions

- SP800-90B Section 4.3 Item 8
  - The submitter shall provide documentation of any known or suspected noise source failure modes (e.g., the noise source starts producing periodic outputs like 101…01), and shall include developer-defined continuous tests to detect those failures. These should include potential failure modes that might be caused by an attack on the device.
- Identify how the source can go wrong, or produce less entropy
  - Environmental conditions, nearby components under heavy load, noise capture device stops receiving signals, changes in voltage

Developer-Defined Health Tests

- SP800-90B health tests are the bare minimum requirements
  - It does not look good on a report when only these tests appear
  - Discuss why these tests cover all possible failure cases for the device
- Failure conditions SHALL be addressed with developer-defined health tests
  - SP800-90B Section 4.3 Item 8
- Provide proof that the concern is mitigated by the test
- A reviewer may ask if the developer considered certain failure conditions

Failure Conditions – Ring Oscillator Example



Failure Conditions – Ring Oscillator Example

- Potential failure: neighboring oscillators could lock together or be interfered with by another signal on the chip
- Oscillators locking together with an XOR would result in diminished entropy
- Address by:
  - Spacing out components and demonstrating the signals provide no interference
    - This can be done empirically with measurements!
  - Add testing to detect if the oscillators lock together
  - Add circuitry to detect if the oscillators lock together

Questions?