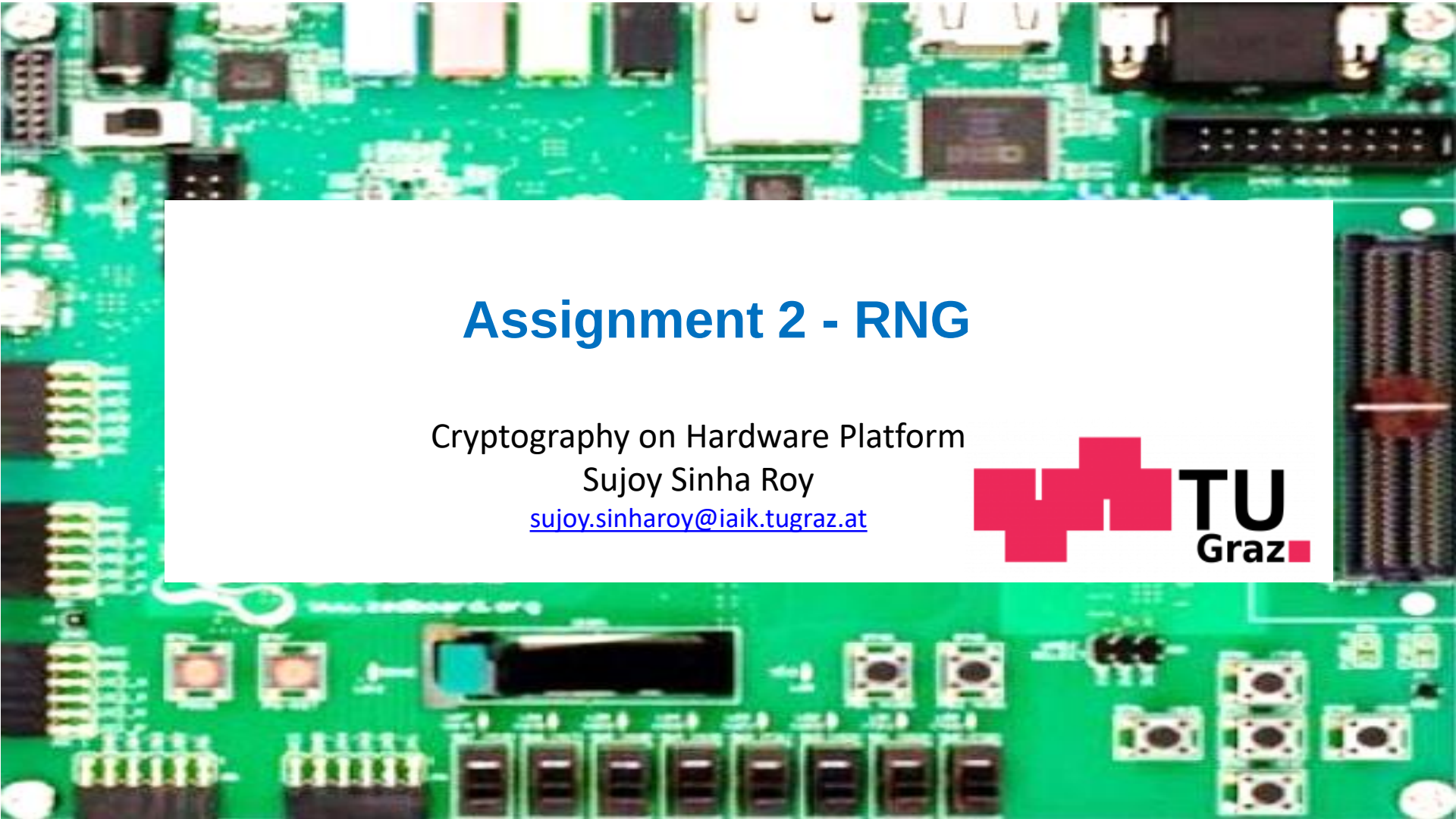


Assignment 2 - RNG

Cryptography on Hardware Platform

Sujoy Sinha Roy
sujoy.sinharoy@iaik.tugraz.at

TU Graz

There are two tasks

Task 1: Design a TRNG in FPGA that meets a high entropy of 0.8 or above.

[Random bits are collected from FPGA by ARM. Next, ARM uses NIST's C++ library to perform statistical tests on the collected data]

Task 2: Perform on-chip statistical testing using "Markov Estimate".

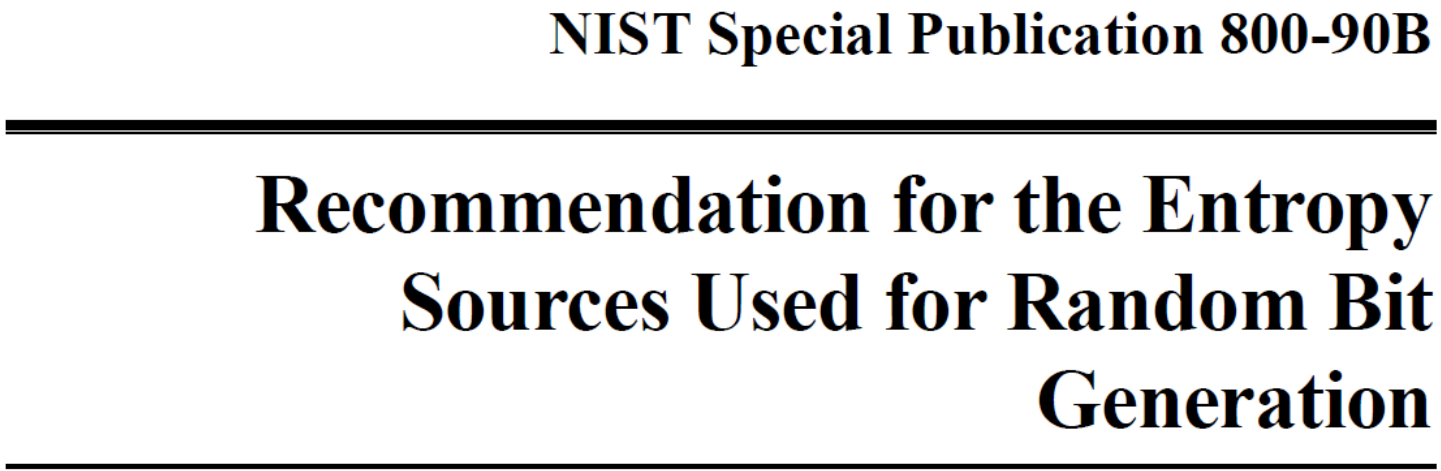
[This test will return an output bit depending on entropy level.
Output bit = 1 when minEntropy <0.8 (Hence TRNG error)
Output bit = 0 when minEntropy >= 0.8 (Hence TRNG is satisfactory)]

Task 1: TRNG design

A reference implementation is provided. It has low entropy.

- Choose appropriate design parameters such that your TRNG meets minEntropy >= 0.8
- TRNG cannot be simulated. You need to do several on-FPGA experimentation to find design(s) with minEntropy >= 0.8.
- Mathematical model may give an indication.
- Use parametric HDL such that you can quickly change design parameters.

Task 2: Markov Estimate



https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-90b.pdf

Page 51 describes Markov Estimate

Task 2: Markov Estimate

Useful hints

Given an N bit string:

1. Number of 0s is C0
2. Number of 1s is C1 = N-C0
3. Number of ‘01’ substrings C01 ≈ Number of ‘10’ substrings C10
4. Number of ‘00’ substrings C00 = C0 – C01
5. Number of ‘11’ substrings C11 = C1 – C10

There are only two independent variables

Task 2: Markov Estimate

Useful tool

Microsoft Excel for linear/polynomial regression analysis