Design, Implementation and Analysis of Efficient Hardware-based Security Primitives

N. Nalla Anandakumar
IIIT-Delhi and University of Florida, USA
nallananth@gmail.com

Introduction

- IoTs market is estimated 75 billion devices are going to be connected to the internet by 2025 in wide-ranging applications.
- With this massive increase, security vulnerabilities are growing exponentially as well
- Therefore, the security of communication between these devices is becoming increasingly important.

Key Security Challenges

- IoTs requires verifying the authenticity of data and identities of devices
  - Establishing the Identity (ID) of each device is very problematic
  - Authentication and attestation use cryptographic protocols that require unique, randomly generated keys for each device.
    * The keys may be long-term keys, session keys for symmetric and public key algorithms.

Traditionally

- IDs of devices/secret keys are stored in non-volatile memories, But (NVM) are often vulnerable to attacks
- In addition, random key generation and key exchange are also very challenging in secure IoT applications
- Hardware-based security primitives such as PUFs and TRNGs to help address above these concerns
  - **PUFs**: used to implement low-cost device authentication, secure secret key or ID generation and key storing
  - **TRNGs**: to provide high entropy random numbers (keys)
  - The IoT infrastructure adopts a large number of these hardware-based security primitives in order to securely exchange data in an effective and resource efficient manner.

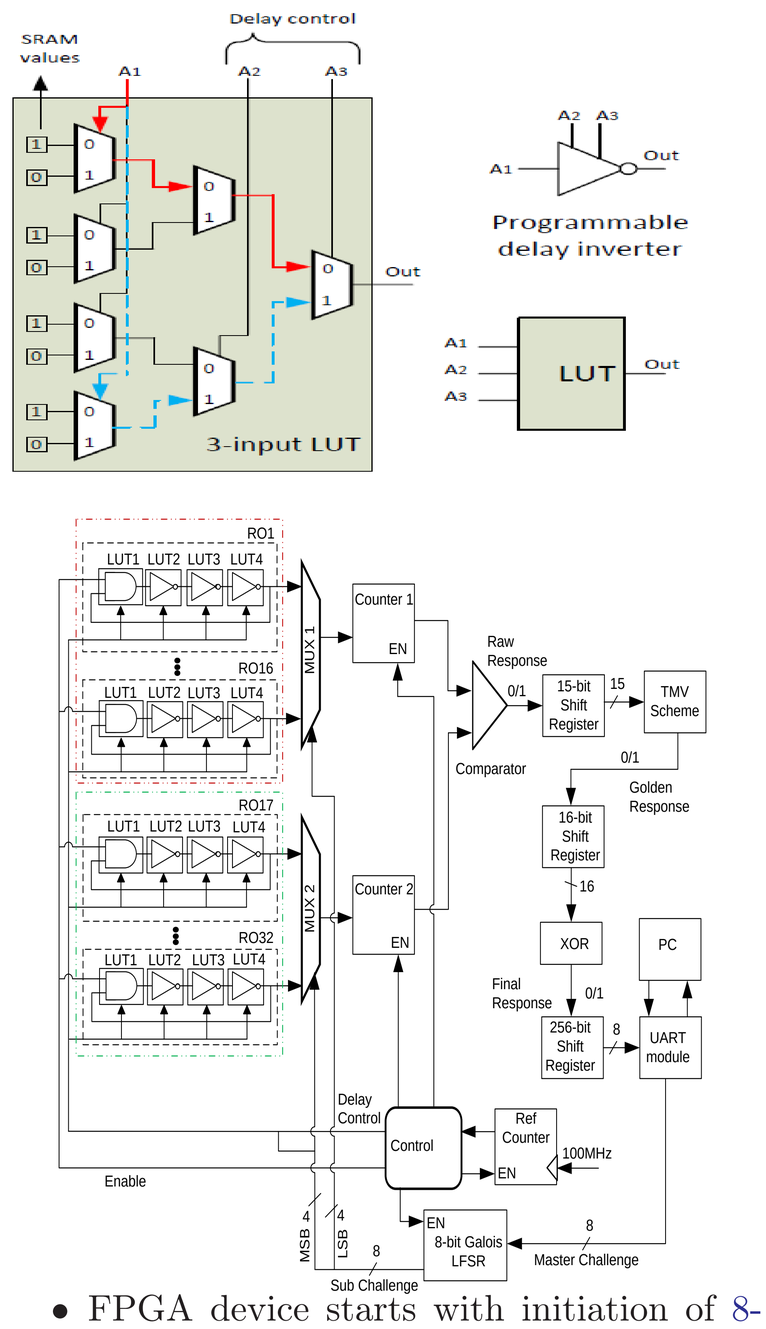
Key Contributions

- At first, we have develop three major types of area efficient PUF designs and improving their qualities.
  1. **Delay** based PUF: Ring oscillator based PUF [1]. (Use differences in wiring delays within the circuit)
  2. **Memory** based PUF [1]: SR-Latch based design. (based on the instability of volatile memory cells)
  3. **Hybrid** PUFs: SR Latch-Arbiter [2]
- In second, we design and develop a ring oscillators based TRNG on FPGA
- In third, we focus on efficient FPGA implementation of elliptic curve based authenticated key agreement protocol for IoT devices using PUF and TRNG.

Conclusion

In this work, we have presented several novel designs and architectures for PUF and TRNG primitives and their practical usability in key agreement protocol. To the best of our knowledge, this is the first implementation of ECMQV using BEC, PUF and TRNG in FPGA till date.

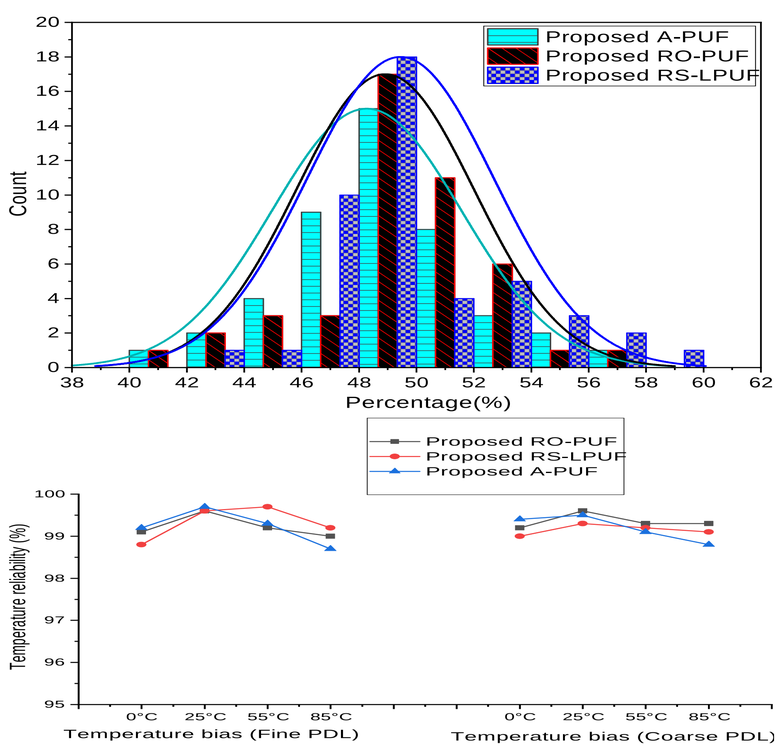
Proposed PUF Design



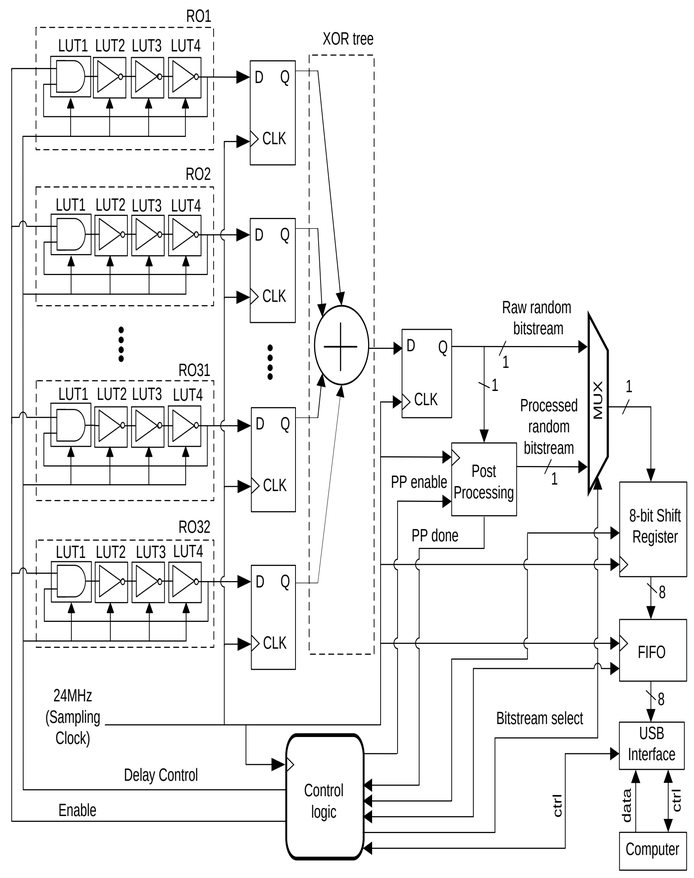
- FPGA device starts with initiation of 8-bit master challenge and generates the 256 subsequent challenges (8-bit LFSR).
- Then from each of these sub-challenges two different ROs are chosen for compari.
- The frequency of the selected ROs are then obtained and fed into the 32-bit counters.
- The comparison of counter 1 and counter 2 values generates a raw resp. bit 0 or 1
- 16 discrete levels (15 times) to the delay control inputs are applied for each subchal
- TMV concept is subsequently applied (golden resp.), and stored in a 16-bit SR
- a 1-bit "final response" is generated by XORing the sixteen 1-bit golden responses

256 response bits are gener. for each 8-bit mc.

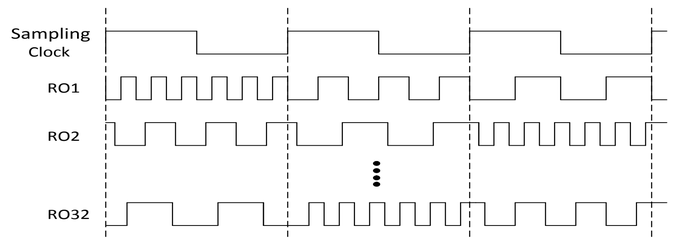
Uniqueness, Reliability



Proposed TRNG Design



- PDL in ROs: to reduce correlation between several equal length oscillator rings, and thus improve the randomness



- The bit sequence passed the 15 NIST, and T8 (AIS31) tests and achieved entropy of 0.9993 per bit [3]).

Key Agreement Protocol

Finally, we presented a practical design for an area efficient authenticated key agreement protocol between two IoT devices using BEC, PUF and TRNG. The key agreement protocol uses PUF for the unique long term secret key generation, TRNG for short term random secret key generation, BEC for generating the public key corresponding to the secret key, and ECMQV for generating the shared secret key [1, 4]. Our implementation shows that the entire protocol can be performed in 151 msec using 15495 slices on a Virtex-5 FPGA [1]

References

[1] N Nalla Anandakumar. *Design, Implementation and Analysis of Efficient Hardware-based Security Primitives*. Theses, IIIT-Dehi, 2020.

[2] N. Nalla Anandakumar, Mohammad S. Hashmi, and Somitra Sanadhya. Efficient and Lightweight FPGA-based Hybrid PUFs with Improved Performance. *Microprocessors and Microsystems*, 77:103180, 2020.

[3] N. Nalla Anandakumar, Somitra K. Sanadhya, and Mohammad S. Hashmi. FPGA-Based True Random Number Generation Using Programmable Delays in Oscillator-Rings. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2019.

[4] N. Nalla Anandakumar, M. Prem Laxman Das, Somitra K. Sanadhya, and Mohammad S. Hashmi. Reconfigurable Hardware Architecture for Authenticated Key Agreement Protocol Over Binary Edwards Curve. *ACM Trans. Reconfigurable Technol. Syst.*, 11(2):12:1–12:19, November 2018.