Properties of improved ROPUF for FPGA generating multiple output bits from each pair of ROs

Filip Kodýtek, Róbert Lórencz, and Jiří Buček
Czech Technical University
Faculty of Information Technology
Prague, Thákurova 9
Email: { kodytfil | lorencz | bucekj }@fit.cvut.cz

Abstract

Many PUF (Physical Unclonable Function) designs for FPGAs proposed up to this day are based on ROs (Ring Oscillators). The classical approach is to compare frequencies of ROs and produce a single output bit from each pair of ROs based on the result of comparison of their frequencies. This ROPUF (Ring Oscillator PUF) design requires all ROs to be mutually symmetric and also the number of pairs of ROs is limited in order to preserve the independency of bits in the PUF response.

Our proposed ROPUF eliminates some of these drawbacks. It is able to generate multiple output bits from each pair of ROs and also allows to pair higher number of ROs. This ROPUF design is easy to implement and is also more effective than the classical approach. The proposed PUF design is based on selecting a particular part of counter value, which is encoded in Gray code and uses it for the PUF output. This counter value is obtained by counting the number of oscillations using 2 counters for each RO in one pair. When one of the counters overflows, the counting is stopped and the value of the second counter is used.

We present the results of measurements that were performed on Digilent Basys 2 FPGA boards. This proposal showed good results when used under stable environmental conditions. On the other hand, this PUF design turned out to be sensitive to the change of voltage.

This behavior led us to further investigate the current PUF design. In this contribution, we provide more detailed description of the PUF design and the behavior of ROs under varying voltage in order to determine the interval of voltage, in which this ROPUF can operate with acceptable error rate. It was verified that the delay in the circuit which detects the overflow and stops the counting does not bring any errors in the form of some offset.