An FPGA Chip Identification Generator using Configurable Ring Oscillator

Mehboob Hasan Ahmed[#1], Rutuja Jagtap[#2], Roopal Pantode[#3,] Prof. S. S. Phule[#4]

[#]*Department Of Electronics and Telecommunication Engineering*
Sinhgad Academy of Engineering, Kondhwa (Bk), Pune-411048, Maharashtra, India

Abstract *— Chip identification has a wide range of applications including Digital Intellectual Property Protection etc. Physically Unclonable Functions (PUF) are commonly used. This can be obtained by various technics, which generates a unique binary string. It is difficult to maintain repeatability of chip ID generation especially over a wide range of operating conditions. To overcome this problem we proposed utilizing configurable ring oscillators and orthogonal re-initialization method to improve repeatability.*

*An implementation on a Xilinx Spartan-3E. FPGA chip was tested. Experimental results show that the bit flip rate is reduced from 1.5% to approximately 0 at a fixed supply voltage and room temperature over a 20-80^0 C temperature range and 25% variation in supply voltage, the bit flip rate is reduced from 1.5% to 3.125 * 10^{-7}.*

Keywords — *Ring oscillator, Physically Unclonable Functions (PUF), FPGA.*

I. INTRODUCTION

Many integrated circuit applications require a unique chip identification number (ID) that can be read anytime during the lifetime of the chip. This trend has led to the techniques in which unique binary strings are associated with the integrated circuits, making it suitable for wide range of applications, ranging from digital rights management, IP protection, integrated circuit counterfeit detection/prevention, cryptographic key generation and device authentication. Field Programmable Gate Array (FPGAs) have a wide range of applications in system on chip applications and embedded system designs and at as main platform to implement other designs. Hence they need to be facilitated with the chip identification capabilities.

The FPGAs these days contains such features. Virtex devices can encrypt the bit stream using a secret key. Downloaded bit stream is decrypted by a hardware decryption core. For this process to work correctly, the device must be programmed with the same secret key which is stored in the RAM [1]. In cool Runner-II CPLD the two technology schemes Dual EDGE and Data GATE confuse attackers with double data rate operation and input signal locking under internal macro cell control [2].

Traditionally, the secret key or the ID information is stored in the volatile or non-volatile memory in an FPGA chip that makes it difficult to be read. Also in SPARTAN-3A "Device DNA" is hardwired into the device which is used to implement designs which only operate with particular ID. These methods can be expensive and increase complexity. Also they are not completely immune to physical attacks. So, instead of stored identification information in a device, Physically Unclonable Function (PUF) that utilizes the physical variation to distinguish chips are used. This concept can be used to obtain chip IDs from mismatch in the delay, voltage or current values of an array of circuit structures of identical designs. These random variations are extracted, averaged and thresholded to produce chip ID.

The chip IDs are required to be unique and repeatable. Unique IDs should be generated for different chips and repeatability is required to make sure that the device returns the same value every time. The chip ID which has low repeatability is referred as unstable. Using Ring Oscillators (ROs) is a method used to generate PUF IDs. Transistors used in this method, create transistor delay mismatched and gives random output from group of ROs with same layout but different spatial locations. This delay is then averaged and thresholded to generate a unique binary string. A run time re-initialization scheme is used to significantly improve the performance.

II.BACKGROUND

PUF's are the innovative circuits which are important for the hardware security research community as they were proposed in 2001 [3],[4]. These enable low cost authentication of individual IC's. A summary of relevant works on various PUF implementation are given below:

A. PUF on ASICs

Lofstrom et al.[5] used an array of addressable NMOS transistors loaded with a common resistive load.

Integrated Circuit Identification (ICID) has been developed that extracts unique and repeatable information from randomness inherent in Silicon processing. This technique can be used with any standard CMOS process. It uses array of addressable MOSFETs with common gate and source. Drain currents are different due to device mismatch, which causes sequence of random of voltages across the load.

By addressing the transistors sequentially, the successive values of voltage generated is connected to a binary sequence to form an unique ID. The device is fabricated using 0.35 micrometer single-poly N-well process. A 112-bit ID is generated with less than 4% drift, less than 10% to 7% error rate with wide range of power supplies, biases, clock frequencies and temperature.

B. PUF on FPGAs

FPGA-based PUF implementations are classified based into following types: Memory-based, Logic-based, Arbiter-based and Ring-oscillator.

- Memory-Based PUF: The SRAM cell can be used to produce the ID bits. It is best known intrinsic PUF based on standard available components. It has been mentioned in Guajardo et. Al that 4% bits vary with time over temperature range 20°C to 80°C [6].

- Logic-Based PUF: Patel et. Al. counted variation-dependent glitches on the output of a combinational multiplier to generate unique identification [8]. They found 6 out of 64 bits are changed over a range of temperature. Anderson used an FPGAs carry chain to implement a PUF [9]. On an average, 3.6% of bits are changed in high temperature.

- Arbiter Based PUF: It consists of two parallel MUX chains feeding a flip-flop. The input transition travels through a series of two input and output switches. Each switch is configured to be either a straight or cross connection based on its selection bit. The difference between the top and bottom path delays is compared by the arbiter and the response bit is generated. Suh and Devadas [10] used this technique and achieved 0.7% unstable bit rate at room temperature.

- RO Based PUF: RO based PUF uses differences in the periods between similar ROs. RO is encapsulated in the hand macro with fixed layout. In comparison with the arbiter-based PUF, the RO based PUF achieves better performance [10]. Maiti and Schaumat [11] proposed configurable RO which achieves higher reliability in RO based PUF. This approach is more efficient in terms of hardware cost. The most common centroid arrangement is used to counter spatial correlations [12].

III.DESIGN

A. *One Bit Generation:*

The bit generation scheme consists of 2x2 RO array to generate one bit. Common centroid layout format is used to place the four ROs as shown in the Figure 1. This arrangement of four ROs is called a "cell" and generates a single bit. An overlapped cell composition is used to improve the resource utilization of the design. 9x9 RO array is required for generation of a 64 bit ID, as compared to 16x16 RO arrays, without compromising the randomness of the generated bits.

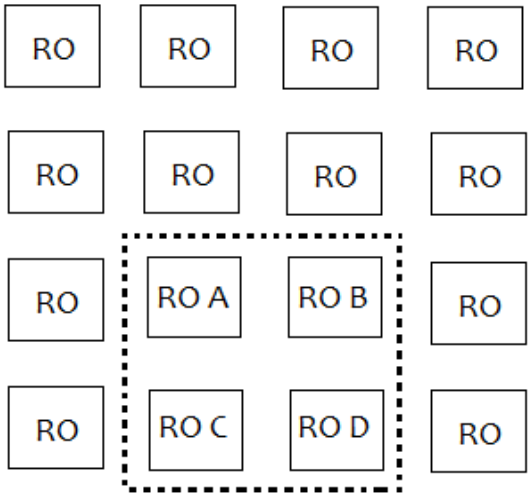


Figure 1. Block diagram of a cell

A timer driven by a 10 MHz system clock, f_{clk} is used to measure the number of rising edges of the RO, NRO, over a period of N timer cycles. The frequency of the RO is hence given by

$$\mathbf{N_{RO} = (f_{RO} / f_{clk}) \times N_{timer}} \quad (1)$$

If N_A, N_B, N_C and N_D are the counter values for the four ROs *A*, *B*, *C* and *D* respectively as shown in figure 2, the residue is calculated as:

$$\mathbf{R_i = (N_A + N_D) - (N_B + N_C)} \quad (2)$$

If R_i is positive, the bit generated by this cell is 0, otherwise, it is 1.

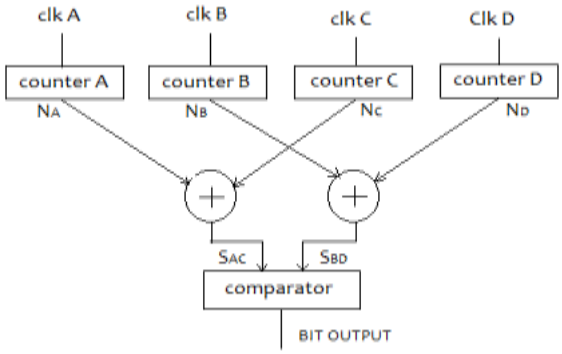


Figure 2. One bit ID generation

If $S_{AC} >= S_{BD}$, Bit = 0 and If $S_{AC} < S_{BD}$, Bit = 1. Static RO has Gaussian distribution; hence most of the residues are near to 0, which makes the system unstable. So, static RO is replaced by configurable RO to increase the residue [13].

IV. IMPLEMENTATION

A. *Architecture*

Figure 3 shows architecture of chip ID generator. The measurement circuitry consists of two main parts. An array of identical ring oscillators and the controller to measure and compare their differential delays. The 9x9 RO array provides 8x8 cells which are arranged in spatially overlapped fashion instead of individual ones. This arrangement serves the significant logic resources and maintains good statistics for ID generation. These cells can generate 64 separate bits (i=0,…..,63). The address generator selects the single RO with the help of two decoders. A 4-bit global RO configuration is sent to each RO. The configuration only affects the operating RO as only one RO can be activated for a given time interval. The FSM controls an internal timer and address generator so that the ID can be generated sequentially. Two levels of MUX route the output of selected RO to the counter. Each RO is connected to input clock of the counter and the number of RO clock periods is measured. The handshaking signals connect the timer to the ARM processor and the residue is calculated according to (2). Handshaking signals include the one-bit enable signal to start the timer operation.

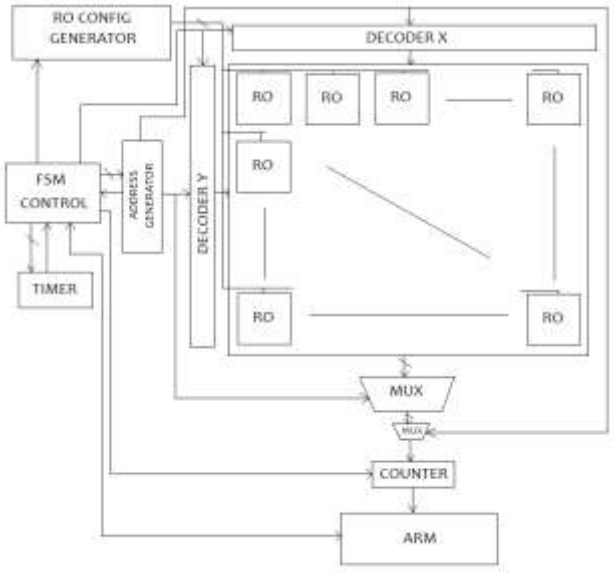


Figure 3. Block diagram of chip ID generator

The post processing is done to facilitate different experiments with ID generator implemented on an external ARM processor in the software.

B. *Configurable RO*

Figure 4 shows configurable RO which is implemented in Xilinx Spartan-3 FPGA. The design can be easily ported to different FPGA families. The 4-stage RO is designed with each stage occupying two Xilinx logic elements (LE) within a

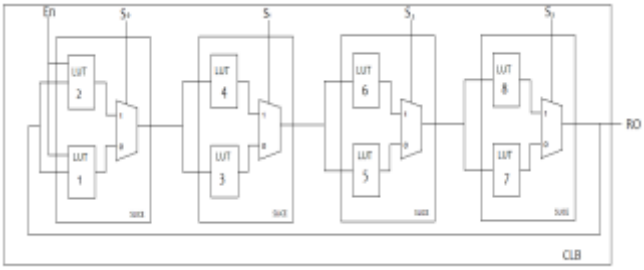


Figure 4. Configurable RO circuit

slice and a multiplexer are used to choose a single path. The entire RO occupies a single Xilinx configurable logic block (CLB). By selecting different values of S0-S3, 16 different configurations can be selected. Logic and Interconnect Delay mismatch in the path changes the frequency of ROs.

C. *Configuration Initialisation*

The bits to be configured need not to be initialised on power-up for generation of stable ID. One approach can be determined by the configurations when FPGA is powered-up for the first time and storing them in non-volatile memory. But, when the digital information is stored in the memory, there can be possibility of information leakage. The

configuration bits is needed to be transferred onto chip for ID generation process via communication channel. Adversaries can extract this information leading to modeling attacks [14].

The other approach gives better results by analyzing three types of cells. Cell #1 produces negative R_i values (negative polarity) for all configurations; Cell #2 produces positive R_i values (positive polarity) and cell #3 produces both positive and negative polarity (hybrid polarity) for all configurations. The configurations with maximum Ri values are selected for ID generation to maximize stability. Hence, the residues over all possible configurations are added, and the sum, S_r is calculated as follows:

- When the residue for all the configurations has the same polarity, then the best configuration among them can be determined by the largest absolute value.
- The polarity of a residue remains unchanged even if the configuration changes with the time. The threshold value can be applied to S_r in case of Hybrid Polarity.
- When it is difficult to determine whether the particular cell has positive or Negative polarity, then divide the configurations into two halves, 0000 to 0111 and 1000 to 1111 and also divide the threshold value. Thus the polarity can be easily identified by checking whether the residue in 1st half is larger or smaller than the new threshold value.

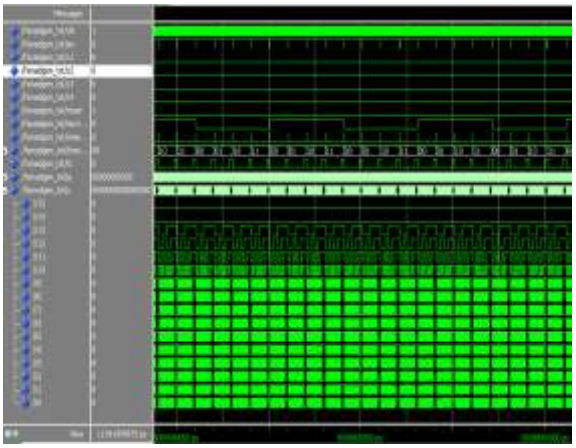
V. RESULT

A. *Hardware Resource Implementation*:

The implementation is done using a custom board with Xilinx Spartan-3A FPGA (xc3s400-8Q208) and an NXP LPC 2148 ARM processor. Xilinx ISE design suite 10.1 is used for FPGA design and µVersion V4 is used for ARM C compilation.

B. Simulation Diagram

The following is the simulation diagram that has been observed and verified in the VHDL Software. It has been observed after successive attempts of Compilation and error rectification.



VI. CONCLUSION

A chip ID generation method using configurable RO can be used to improve repeatability and stability. These parameters can be improved by power up initialization and an adaptive re-initialization. Minimum resources are utilized in implementing this design. This design can also be implemented in an ASIC as standard digital circuits are used to implement this design. As future work, more schemes can be generated to speed up chip ID generation.

REFERENCES

[1] A. Telikepalli, "Is your FPGA design secure?" *Xilinx XCELL*, vol. Fall, 2003.

[2] KavitaC.Mugali, MinaksheeM.Patil "Configurable Ring Oscillator for FPGA chip Identification"; Technovision-2014:1st International Conference at SITS, Narhe, Pune on April 5-6, 2014.

[3] R. Pappu, "Physical one-way functions," Ph.D. dissertation, Massachusetts Institute of Technology, 2001.

[4] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical One-Way Functions," Science, vol. 297, no. 5589, pp. 2026–2030, 2002.

[5] K. Lofstrom, W. R. Daasch, and D. Taylor, "IC identification circuit using device mismatch," in Proceedings of the International Solid State Circuits Conference (ISSCC), 2000, pp. 372–373.

[6] J. Guajardo, S. Kumar, G.-J. Schrijen, and P. Tuyls, "Physical unclonable functions and public-key crypto for FPGA IP protection," in Field Programmable Logic and Applications, 2007. FPL 2007. International Conference on, aug. 2007, pp. 189 –195.

[7] H. Patel, Y. Kim, J. McDonald, and L. Starman, "Increasing stability and distinguishability of the digital fingerprint in FPGAs through input word analysis," in Field Programmable Logic and Applications, 2009. FPL 2009. International Conference on, September 2009, pp. 391 –396.

[8] J. Anderson, "A PUF design for secure FPGA-based embedded systems," in Design Automation Conference (ASP-DAC), 2010 15th Asia and South Pacific, jan. 2010, pp. 1 –6.

[9] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in DAC '07: Proceedings of the 44th annual Design Automation Conference. NewYork,NY,USA: ACM, 2007, pp. 9–14.

[10] A. Maiti and P. Schaumont, "Improving the quality of a physical unclonable function using configurable ring oscillators," in Field Programmable Logic and Applications, 2009. FPL 2009. International Conference on, 31 2009-sept. 2 2009, pp. 703 –707.

[11] H. Yu, P. Leong, H. Hinkelmann, L. Moller, M. Glesner, and P. Zipf, "Towards a unique FPGA-based identification circuit

using process variations," in Field Programmable Logic and Applications, 2009. FPL2009. International Conference on, 31 2009-Sept. 2 2009, pp. 397–402.

[12] T. W. Anderson and D. A. Darling, "Asymptotic theory of certain
"goodness of fit" criteria based on stochastic processes," in Ann. Math. Statist., vol. Volume 23, 1952, pp. 193–212.