Dynamic Programming Based Grouping Method for RO-PUFs

Giray Kömürcü
National Research Institute of Electronics and Cryptology,
TÜBİTAK, 41470, Kocaeli, TURKEY
Email: girayk@uekae.tubitak.gov.tr

Ali Emre Pusane, Günhan Dündar
Bogazici University, Dept. of Electrical and Electronics Eng.
34342 Bebek, Istanbul, Turkey
Email: {ali.pusane, dundar}@boun.edu.tr

***Abstract*—Key generation is one of the most promising applications of Physical Unclonable Functions (PUFs), which requires 100% robust bit streams within each circuit and true randomness among a set of circuits. However, due to the noisy nature of PUFs, it is hard to provide stable outputs under changing environmental conditions, such as supply voltage and temperature. In this work, we have adapted Dynamic Programming (DP) to RO-PUFs for the first time in literature, in order to extract maximum entropy with minimum possible resource usage. Next, the robustness of all output bits is guaranteed even in unstable environmental conditions just by measuring a small subset of circuits prior to shipment. Finally, the efficiency of our method is analyzed and validated experimentally with FPGA implementation.**

***Keywords*-PUF, Physical Unclonable Functions, Reliability, Robustness, Ring Oscillator, Dynamic Programming, FPGA.**

I. Introduction

Protection of cryptographic keys is the most important issue in security operations. In conventional systems that have no constant power sources such as smart cards, keys are either transferred once and stored in non-volatile memories, or transferred to the device whenever needed. Secure transfer of keys is problematic in both situations, since it requires implementation of complex protocols to protect the keys against snooping, and non-volatile memory usage is not suitable for resource limited devices.

Physical Unclonable Functions (PUFs) offer promising solutions in the area of key generation and storage. These functions, which have the unique capability of generating chip specific signatures on the fly, were first introduced in 2001 [1]. Among various PUF types presented in the literature such as Arbiter PUFs, SRAM PUFs, Butterfly PUFs, and Glitch PUFs [2]–[7], RO-PUFs are the most convenient type for FPGA implementation [8] and exhibit higher reliability than other PUF types under changing environmental conditions [9]. This work focuses on building a 100% reliable and highly efficient PUF structure based on ROs suitable for cryptographic key generation.

Output generation mechanism of conventional RO-PUFs depends on the pairing approach, which generates one bit output with a pair of ROs [10]. To extract the maximum entropy from the system, frequency ordering of all ROs have to be used, which can generate up to $\lfloor \log_2(N!) \rfloor$ bits by using *N* ROs [10]. Even though this theoretical upper-bound is not achievable due to noise in the system that is causing unreliable bits, it is still much higher than the number of bits generated in conventional systems, which is upper-bounded by *N/2*. The frequency ordering approach is used in [11] for output generation, which is called the Longest Increasing Subsequence-Based Grouping Algorithm (LISA) .

In this work, DP algorithm is adapted to RO-PUF output generation for achieving maximum entropy with minimum resources. Secondly, we introduce a parameter called predetermined frequency threshold (f_{thp}) to be used in DP for a 100% reliable PUF. Finally, the effectiveness of the proposed algorithm in terms of entropy generated, computational cost, and area utilization is analyzed. The rest of the paper is organized as follows: In Section 2, background for RO-PUFs is provided and methods of maximizing the entropy are discussed. In Section 3, DP is adapted to RO-PUF output generation for maximum entropy with highest robustness in minimum time. In section 4, experimental validation is performed. Last section concludes the paper.

II. Maximizing Entropy and Robustness in RO-PUF Circuits

RO-PUFs are structures that generate output depending on the frequency differences of identically laid out ROs. For continuous oscillation, an odd number of inverting delay stages are connected serially forming a ring structure. In vast majority of RO-PUFs, one bit output is generated by comparing the frequencies of two ROs [12]–[14]. In such systems, the area of the implementation is usually high due to the low entropy utilization by the structure [11].

To overcome the entropy extraction problem, comparing more than two ROs at a time is required. The first step is grouping of the ROs that will be compared at once. The frequencies of ROs in each group should be adequately seperated from each other, preventing changes in frequency ordering due to noise in the system and temperature or supply voltage fluctuations. For a group of *M* ROs, *M!* different orderings that are equally likely may occur. By mapping each different ordering to a bit stream, $\lfloor \log_2(M!) \rfloor$ bits can be generated from each group [10]. In this approach the main problem is to form the largest possible groups from the set of ROs implemented in the circuit.

In the literature, LISA is used to overcome the grouping problem [11]. In LISA, noise in the system is compensated with the f_{th} value and the effect of environmental changes are compensated by measuring each RO in two extreme conditions and using both values in the algorithm. Even though this approach guarantees robustness, it is quite complex, since it requires two measurements for all ROs in all circuits. Also, the computation cost increases since two frequencies are used per RO in the algorithm. In our proposed method, the frequency deviation that may result from temperature and supply voltage changes are compensated by including them in the f_{th} in LISA and this new parameter is called f_{thp}. Therefore, the value of f_{thp} will be higher in DP. With this approach ROs are measured once under normal operating conditions and DP works with only one frequency value per RO.

The key point in this new approach is determining the value of f_{thp}. If the value determined is less than the required amount, bigger groups will be formed, but their RO frequencies will not be far enough from each other. In this case, ordering may change under certain conditions preventing the PUF outputs from being 100% reliable. On the other hand, if the value of f_{thp} is higher than the required amount, smaller but more reliable groups will be formed and the extracted entropy, hence the number of output bits generated, will be lower. In order to determine the optimum f_{thp}, a small subset of circuits is formed and the frequencies of the ROs in this subset is measured at two different temperatures. Normally, all ROs will be slower at higher temperatures, but their frequency change will be different from each other. This difference is the reason for unreliable bits and should be used as minimum f_{thp} in the PUF output generation algorithms. A formal structure of determining the f_{thp} is given in Algorithm 1.

```
Data:
1. A list of minimum RO frequencies $fmin[n]$.
2. A list of maximum RO frequencies, $fmax[n]$.
Result: $f_{thp}$
for $i \leftarrow 1$ to $n$ do
    $diff(i) = fmax(i) - fmin(i)$;
end
$f_{thp}$=$max(diff) - min(diff)$;
```

Algorithm 1: Determining f_{thp} in pseudo code

With the proposed method, a realistic value of f_{thp} is determined. Since a subset of all circuits is used, a small amount of overhead should be added to this value for guaranteed robustness in all manufactured circuits.

III. Adapting dynamic programming to RO-PUFs

Even though LISA extracts the maximum available entropy from the system with guaranteed robustness, it is very costly in terms of computational power, mainly due to redundant search for ROs to form the optimum groups. By using LISA, it may be hard to achieve low computation times for output generation on devices with limited capability. In addition to this, it requires two measurements for each circuit at two extreme temperatures which complicates and increases the cost of initialization and output generation phases.

To overcome the drawbacks of LISA without decreasing its capability of extracting entropy and achieving high robustness levels, we have adapted DP to this problem. With this approach, the computational complexity of output generation decreases considerably and the requirement to measure each circuit at two extreme temperatures during the initialization phase is avoided. Measuring a small subset of circuits at extreme conditions is enough to determine f_{thp}, which will guarantee reliability with just one measurement for the rest of the circuits.

The inputs to the DP for RO grouping is similar to the inputs of LISA. Each RO frequency is measured during the initialization phase and given as input to the algorithm. In addition to this, f_{thp} is also used by the algorithm for generation of reliable outputs. By using this f_{thp} parameter, it is possible to avoid measuring and working on two different frequencies for each RO, reducing the complexity of the initialization phase and PUF output generation.

DP algorithm has three steps. In the first step, ROs are sorted according to their frequencies in increasing order and $Fsorted[n]$ list is created. In the second step, for each RO, the nearest RO whose frequency is at least f_{thp} higher is found and a linked list is created, $list[n]$. In the third step, groups are formed using the linked list, that satisfy the requirements of maximum entropy and 100% robustness even in unstable environmental conditions. In this step, the algorithm starts from the first position in the list, $list[1]$, and groups RO_1 with the RO that $list[1]$ shows, RO_j. Then, DP continues by jumping to the position in $list$ of last grouped RO, j and group the one that $list[j]$ shows. This continues until last position is reached in $list$. After the first run, the first group is formed and this step is repeated until all ROs are grouped. During the grouping process, if the RO that the list shows is grouped, the nearest ungrouped RO through the end of list is added to the group. DP algorithm is illustrated with a small data set in Figure 1. In this figure, $FreqRO[n]$ represents the frequencies of ROs in MHz. Also, f_{thp} is selected as 1.5 MHz. After the algorithm is applied, 3 distinct groups are formed satisfying the reliable PUF output generation conditions with maximum entropy extraction. The pseudo code of the DP approach is presented in Algorithm 2.

The reduced complexity of the proposed DP approach is a result of avoiding the redundant RO search done by the LISA. For each addition to a group, LISA searches over all ROs that have at least f_{th} higher frequency than the last group member. Once a possible RO is identified, it is added to the group. In the DP, benefiting from the fact that we are operating on a sorted RO frequency list, the first qualifying candidate is chosen. This simplifies the task, since it amounts to choosing the nearest item to the last group member in the sorted RO frequency list. This search is illustrated in Figure 2, where the last group member is assumed to be i. The LISA algorithm searches over the region of the remaining sorted RO list, where there is at least f_{th} frequency difference, whereas the DP approach simply chooses i''. Choosing the first available candidate seems like a suboptimal solution; however, as we

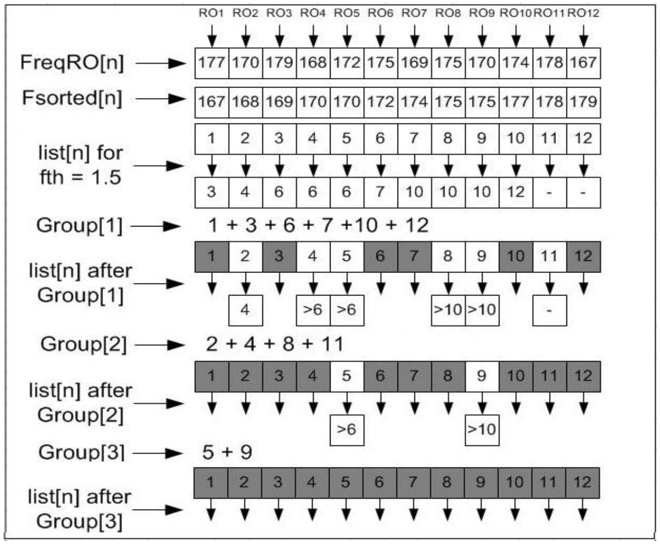

Fig. 1. DP sample execution for 12 elements.

```
Data:
1. A linked list of ROs with their frequencies measured
under normal operating conditions, $FreqRO[n]$.
2. $f_{thp}$ for robustness
Result: Groups of ROs.
Sort $FreqRO[n]$ by frequency in increasing order:
$Fsorted[n]$
for $i \leftarrow 1$ to $n-1$ do
    find the nearest element $Fsorted[j]$ that is
    ($Fsorted[i] < Fsorted[j]$-$f_{thp}$) and link $i$ to $j$ in
    $list[n]$
end
$i = 1$
while ungrouped RO exists do
    if $ROi$ is ungrouped then
        Add $ROj$ to the group of $ROi$
        Jump to $ROj(i = j)$
    end
    if $ROi$ is grouped then
        Increment $i$ until $ROi$ is ungrouped
    end
    if i=n and still ungrouped RO exists then
        $i = 1$
    end
end
```

Algorithm 2: Dynamic Programming in pseudo code

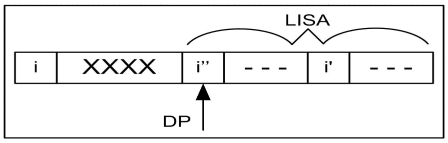

Fig. 2. Search for largest group

TABLE I
MAXIMUM FREQ. DEVIATION OF ROS DUE TO TEMP. CHANGE

Initialization Temperature (^{o}C)	Min./Max. Operation Temp. (^{o}C)	Max. Frequency Deviation (kHz)
20	0	296
20	40	242
20	60	362
20	80	661
20	100	985

will prove next, this approach is indeed optimal in the sense that it always adds the same group member as LISA, thanks to the sorted nature of the input RO frequency list. We attempt to prove this via proof by contradiction.

Let $\{S_i\}$ denote the largest group that starts at position i (and ends at position n) and g_i denote the size of this group, $1 \leq i \leq n$. Also note that f_i denotes the frequency for the corresponding RO. The LISA algorithm searches over all possible future positions to obtain i' that belongs to the largest group, i.e., $i' = \arg\max(g_{i'})$ for all $i' > i$, such that $f_{i'} - f_i > f_{th}$. In this case, we can form the largest group for position i, by simply adding it to this group:

$$g_i = 1 + g_{i'}, \quad (1)$$
$$S_i = S_{i'} \cup \{i\}. \quad (2)$$

On the other hand, the DP approach simply looks for the smallest i'' such that the f_{thp} condition is satisfied using $i'' = \arg\min(i'')$ for all $i'' > i$, such that $f_{i''} - f_i > f_{thp}$. The corresponding group can be formed by

$$g_i = 1 + g_{i''}, \quad (3)$$
$$S_i = S_{i''} \cup \{i\}. \quad (4)$$

Our claim is that the newly formed group using the DP approach is at least as large as that, formed by the LISA algorithm, i.e., $g_{i''} \geq g_{i'}$. Now, let's assume this is incorrect, i.e., assume that $g_{i''} < g_{i'}$. In this case, we can simply take $S_{i'}$ and replace i' with i'' using $S_{i''} = S_{i'} \setminus \{i'\} \cup \{i''\}$. This is indeed a valid set, since $f_{i''} \leq f_{i'}$ and any group that has the position i' as its lowest value would be still valid if this change is completed. This leads to the fact that we have a group that starts at position i'' and has the same number of elements as that of $S_{i'}$, i.e., $g_{i''} = g_{i'}$. This is a contradiction, since we had started with assuming $g_{i''} < g_{i'}$. Hence, the claim of $g_{i''} \geq g_{i'}$ is valid and the DP can indeed form groups that are at least as large as the ones formed by the LISA.

IV. EXPERIMENTAL ANALYSIS AND VALIDATION

In the system we have set up, 160 ROs are placed on a Xilinx 3S5000 chip and their frequencies are sent to PC via MATLAB. The frequencies of each RO are measured at 6 different temperatures, 0^{o}C, 20^{o}C, 40^{o}C, 60^{o}C, 80^{o}C, 100^{o}C to be able to calculate the related f_{thp} values under different environmental conditions. It is assumed that the initialization of the PUF circuit is done at 20C^o and all f_{thp} values are calculated with reference to the frequencies measured at this temperature which are given in Table 1.

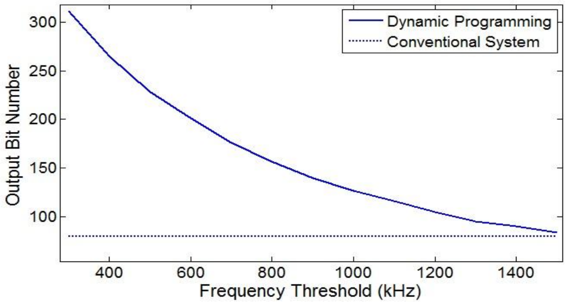

Fig. 3. PUF output bit generation comparison.

TABLE II
AREA REDUCTION WITH DP

f_{thp} **(kHz)**	**Num. of ROs for 80 bit out with Dyn. Prog.**	**RO num. decrease** (%)
600	75	53
1000	105	34
1200	122	23

From this point on, we have analyzed the effectiveness of DP for different values of f_{thp} in a wide range, since different designs may require different operating regimes, hence different f_{thp} values. In the real case, it is the designers responsibility to determine the correct temperature and/or supply voltage for the reference and extreme measurement cases for an effective f_{thp} determination.

In one-by-one comparison based systems N ROs can generate $N/2$ bits without any dependency. In ordering based systems, theoretical upper-bound is $\lfloor \log_2(N!) \rfloor$. By using 160 ROs, 80 bit output can be generated with conventional systems. On the other hand, in ordering based systems, the upper-bound is 1086 bits which is more than 13 times higher. But this is not achievable due to the noise in the system and changing environmental conditions. When f_{thp} is added to compensate these effects, the generated number of output bits decreases. For instance, by using 1000kHz f_{thp}, DP generates 127 bits of output which is significantly higher than the 80 bits of one-by-one comparison systems. Moreover these conventional systems do not guarantee 100% robustness. Number of bits generated by DP with respect to conventional systems by using different f_{thp} values are shown in Figure 3.Since more entropy is extracted from the system with ordering based output generation mechanisms, less RO implementation is enough for the same number of output bits. In Table 2, required number of ROs for DP algorithm is presented for different f_{thp} values. As seen from the table by using 600kHz as f_{thp}, more than 50% area reduction is achieved.

Our another aim was to decrease the computational cost of the output generation in ordering based systems. To analyze the computational cost of mentioned methods, both algorithms are implemented on MATLAB and computation times are measured under the same conditions. As seen in Figure 4, DP has significant advantage over LISA. For instance, for 160 RO implementation, the execution time of LISA is more than 5.5 times longer than the execution time of DP.

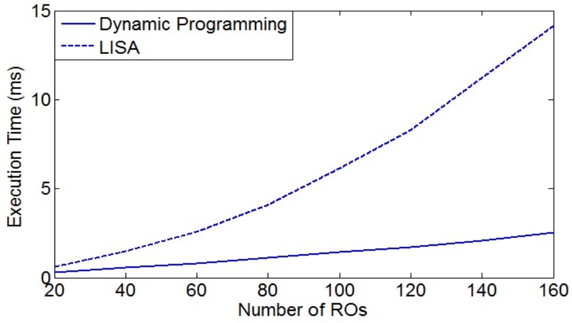

Fig. 4. Execution time of algorithms in MATLAB.

V. CONCLUSION

We have adapted Dynamic Programming approach to PUF output generation in order to maximize entropy extraction from a certain number of ROs with a minimum computational complexity for the first time in the literature. In addition to this, a method for achieving 100% robustness is proposed. Lastly, the efficiency of proposed methods is analyzed and validated experimentally with FPGA implementation.

REFERENCES

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

[2] D. Lim, J. Lee, B. Gasend, G.E.Suh, M. V. Dijk, and S. Devadas, "Extracting secret keys from integrated circuits," *IEEE Transactions on VLSI Systems*, vol. 13, no. 10, pp. 1200–1205, 2005.

[3] B. Gassend, D. Clarke, M. V. Dijk, and S. Devadas, "Delay-based circuit authentication and applications," in *ACM Symposium on Applied Computing*, 2003, pp. 294–301.

[4] B. Gassend, "Physical random functions," M.S. Thesis, Massachusetts Institute of Technology, 2003.

[5] J. Guajardo, S. Kumar, G. Schrijen, and P. Tuyls, "FPGA intrinsic PUFs and their use for IP protection," in *18th Annual Computer Security Applications Conference (CHES)*, vol. 4727, 2007, pp. 63–80.

[6] J. Guajardo, S. Kumar, R. Maes, G. Schrijen, and P. Tuyls, "Extended abstract: The butterfly PUF protecting IP on every FPGA," in *Hardware-Oriented Security and Trust (HOST)*, 2008, pp. 67–70.

[7] D. Suzuki and K. Shimizu, "The glitch PUF: A new delay-PUF architecture exploiting glitch shapes," in *Cryptographic Hardware and Embedded Systems (CHES)*, 2010, pp. 366–382.

[8] A. Maiti and P. Schaumont, "Improved ring oscillator PUF: An FPGA-friendly secure primitive," *Journal of Cryptology*, vol. 24, no. 2, pp. 375–397, 2011.

[9] C. Yin and G. Qu, "Temperature aware cooperative ring oscillator PUF," in *IEEE International Workshop on Hardware Oriented Security and Trust (HOST)*, 2009, pp. 36–42.

[10] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," *Design Automation Conference (DAC)*, pp. 9–14, 2007.

[11] C. Yin and G. Qu, "LISA: Maximizing RO-PUF's secret extraction," pp. 100–105, 2010.

[12] A. Maiti and P. Schaumont, "Improving the quality of a physical unclonable function using configurable ring oscillators," *International Conference on Field Programmable Logic and Applications, (FPL)*, pp. 703 –707, 2009.

[13] H. Yu, P. Leong, H. Hinkelmann, L. Moller, M. Glesner, and P. Zipf, "Towards a unique FPGA-based identification circuit using process variations," *International Conference on Field Programmable Logic and Applications (FPLA)*, pp. 397–402, 2009.

[14] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Silicon pysical random functions," in *ACM Conference on Computer and Communications Security (CCS)*, 2002, pp. 148–160.