A Low-power APUF-based Environmental Abnormality Detection Framework

Hongxiang Gu, Teng Xu and Miodrag Potkonjak
Computer Science Department
University of California, Los Angeles
Los Angeles, California 90095
Email: {hxgu, xuteng, miodrag}@cs.ucla.edu

***Abstract*—Physical unclonable functions (PUFs) take advantage of the effect of process variation on hardware to obtain their unclonability. Traditional PUF design only focuses on the analog signals of circuits. An arbiter PUF, for example, generates responses by racing delay signals. Implementations of such PUFs usually employ large area and power consumption while providing very low throughput.**

To address this problem, we propose an energy efficient PUF design in such a way that it races analog signals and computes digital logic simultaneously. More importantly, the analog portion of the circuit (racing) shares a large amount of hardware resources with the digital portion of the circuit (computing) by introducing only small overhead in terms of area and power. Our test results on Spartan-6 field-programmable gate array (FPGA) platforms indicate that by combining the two outputs, our design enables much larger PUF output throughput, better randomness and less power consumption compared to traditional PUFs.

I. Introduction

The massive deployment of mobile reconfigurable devices has imposed a high-reliability requirement. It has become necessary that these devices should operate reliably irrespective of the change in the external operating environment. In order to monitor the reliability of these mobile reconfigurable devices, measurement of physical operating parameters like on-chip power supply voltages and die temperatures is desired in many situations. An alarm should be raised whenever the reliability is compromised due to abnormal or extreme operating environment. Environmental abnormalities are dangerous because, on the one hand, sudden change in the environment could indicate faulty hardware (circuit short) or logic failure (excessive switching); on the other hand, abrupt variation could also be caused by a variety of physical attack methods, putting the integrity of the operation and the intellectual property embedded in the hardware at the fringe of danger. In both cases, notification of such changes is desired to avoid further damage. Conventionally, detection of abnormal variations requires on-chip temperature and voltage sensors. However, sensor-based monitor system has its drawbacks such as relatively high power consumption and low sample rate. As a matter of fact, many low-end reconfigurable devices like Internet of things(IoT) nodes do not have such sensors embeded. Thus, a low cost, high accuracy abnormal environment detection method is desired to monitor mobile devices without the help of temperature and voltage sensors.

978-1-5090-6023-8/17/$31.00 ©2017 IEEE

APUF, a unique type of hardware security primitive that has been used in many security applications, has excellent properties include low power and high speed. However, APUFs suffer from the notorious reputation of instability and environmental sensitivity. Challenge-response mapping for a specific APUF is usually not stable: the mapping changes as environmental factors varies. Slight change in temperature and voltage could lead to the unpredictable remapping of many CRP. Several APUF-based security applications intend to minimize the impact introduced by instability using stable CRP selection or error correction code (ECC). While considerable efforts are seeking to eliminate the problem, we see the environmentally sensitive nature of an APUF as an opportunity. The instability due to environmental changes can be exploited to detect abnormalities during the operation cycle of a protected hardware.

We propose to use an APUF as an environmental abnormality detector. The core design idea is based on the observation that changes in an APUF's challenge-response mapping imply a highly probable variation in operating voltage or temperature. We propose to monitor the remapping activity of only ESCRPs instead to save additional energy and latency. Our design is advantageous comparing to current sensor-based system monitors for the following three reasons:

- Implementation is flexible. Detection of environmental variation is not a must for all applications. Our detection framework can be implemented or removed easily on reconfigurable hardware depending on user's demand.
- High sample frequency. Many abnormalities only affect the device for a few clock cycles, current system monitor sample rates on Xilinx Virtex devices (200kHz maximum) fails to detect these variations. Our APUF-based framework is capable of sampling at a much higher frequency (around 10 Mhz in our implementation).
- Low power and compact size. Our detection framework requires only a single APUF to detect abnormalities, which grants us a huge advantage in terms of area and power. After careful off-line ESC seed generation and alarm threshold calibrations, the system provides comparable detection rates with state-of-the-art detection mechanism.

II. Related Work

A. Physical Unclonable Functions

PUF was first proposed by Pappu et al. using mesoscopic optical systems [1]. Gassend et al. developed the first silicon PUF through the use of intrinsic process variation in deep submicron integrated circuits [2]. More recently, efforts have been made to enhance the security and randomness of PUFs. A robust PUF design is proposed by Xu et al. with a selected PUF challenge-response set [3]. A low power high randomness PUF design was proposed by Gu et al. by combining classic PUFs with hardware random number generators [4]. Xu et al. also proposed to manipulate programmable delay lines to increase the PUF stability [5]. All past PUF studies in the literature intended to reduce environmental impact on PUFs. To the best of our knowledge, we are the first to propose a mechanism that utilizes PUF's environmental sensitivity as a tool.

B. System Monitor

Many efforts have been made to monitor the physical parameters of a reconfigurable system. Xilinx has embedded their System Monitor in their recent FPGA products with relatively low sample rate [6]. Le Masle and Luk introduced a ring oscillator-based attack detection system that monitors the core power of a circuit through observing the behavior of embedded ring-oscillators [7]. Even though the ring oscillator-based power monitor framework is capable of detecting abnormalities at a much higher sample rate, it is also much larger in size. Despite the existing studies, our proposed APUF-based abnormality detection framework is capable of achieving high sample rate while remaining simple, compact and low power.

III. Preliminaries

A. APUF Model

The PUF we use in our abnormality detection framework is standard delay-based APUFs. An n-bit APUF takes an n-bit challenge as input and produces a 1-bit response as output. When an n-bit challenge is provided to the APUF, two identically designed paths are generated along a chain of multiplexer pairs. Each challenge bit controls whether the pair of paths should swap positions. A pair of multiplexers is denoted as an APUF segment. To retrieve a response, an impulse signal is fed into the system to excite both paths simultaneously. Because of the uncontrollable process variation, the signal traveling along one of the two paths will

Assuming an APUF maps challenges C to corresponding responses R. The APUF has n segments meaning all challenges C are n-bit long. We assume no delays on the connection wires and all delays are contributed by the APUF segments. Given a specific challenge $c \in C$, the ith APUF segments generates a pair of delays with delay difference of Δd_i^c. In a stable environment, the corresponding response $r \in R$ can be mathematically represented as:

$$r = \begin{cases} 0 & \text{if } \sum_{i=1}^{n} \Delta d_i^c > 0 \\ 1 & \text{if } \sum_{i=1}^{n} \Delta d_i^c < 0 \end{cases} \tag{1}$$

In real life, however, environmental variations change the delay difference in each APUF segment. For simplicity purposes, we assume that a minor change in environmental would change the delay difference in the ith segment by Δd_i^e where $\Delta d_i^e \in [-e, e]$, e is the maximum change each delay difference could be altered in a normal operational environment. Δd_i^e is a function of temperature t and voltage v. Due to on-chip temperature and voltage gradient, t, v are different from segment to segment. When provided with a specific challenge $c \in C$, the response $r \in R$ can be mathematically represented by

$$r = \begin{cases} 0 & \text{if } \sum_{i=1}^{n} \Delta d_i^c + \Delta d_i^e > 0 \\ 1 & \text{if } \sum_{i=1}^{n} \Delta d_i^c + \Delta d_i^e < 0 \end{cases} \tag{2}$$

Noted that our abnormality detection framework is not limited to only standard APUF, many other delay-based PUFs like ring oscillator PUFs serve our need as well. However, APUF is more environmentally sensitive, more compact and low power comparing to ring oscillator-based PUFs, so in this work, our design and implementation are entirely based on the above APUF structure.

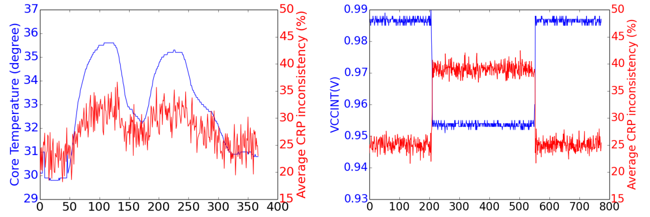
IV. CRP Environmental Sensitivity

Physical properties of transistors vary as environment changes, thus a large number of APUF CRPs are sensitive to fluctuations in physical parameters. In this section, we first discuss correlations between CRP consistencies and physical parameters in the operating environment. We then present a method to collect a set of ESCRP with strong correlation with environmental parameters. We intend to show that the CRP inconsistency in APUFs can serve as a good indicator of environment variations. We define **CRP inconsistency** as the probability for a corresponding response being inverted when the challenge remains unchanged.

A. Environmental Variation

Operating environment is a general concept that includes all on and off chip physical conditions during the process of an operation. Humidity, room temperature, supply power all play an important role during the process of logic operations on a reconfigurable device. In this paper, we mainly focus on studying on-chip voltage and temperature. We carefully examine the core temperature and voltage levels in a reconfigurable platform and their relationship with the on-chip APUF CRP inconsistency.

1) Core Temperature vs. CRP inconsistency: Core temperature can be fluctuated by many factors including but not limited to circuit switching activity, room temperature, heat sink efficiency etc. A sudden, unintended and significant increase or decrease in the core temperature is defined as **temperature abnormality**. Temperature abnormalities could result in circuit behavior alternation, hardware malfunction or even physical damage. APUF serves as an excellent temperature abnormality detector because transistor delays are sensitive to temperature. Change in transistor delays potentially invert the signal racing results between two delay paths in an APUF.



(a) Core temperature(°C) vs. average CRP inconsistency(%) for 100 randomly selected challenges.

(b) VCCINT(V) vs. average APUF CRP inconsistency(%) for for a set of 100 randomly selected challenges.

Fig. 1: A snippet of core environment change vs. inconsistency of randomly selected APUF CRPs. The y-axis on the left is core temperature corresponding to the blue line, the y-axis on the right is the average CRP inconsistency corresponding to the red line.

We first randomly selected 1,000,000 64-bit binary strings as our test challenge set. Each challenge within the test challenge set is repeatedly fed to the 64-bit APUF implemented on a Virtex-5 FPGA. We adopt the majority voting scheme to decide on a standard response. For a challenge C, if a majority of the corresponding responses is R, the tuple (C, R) is recorded as the reference CRP. All reference CRPs are stored in a dictionary $Dict_{CRP}$. Core temperature is being sampled and recorded simultaneously using on-chip sensors. A set of 100 challenges were being evaluated between every two sensor samples. Each set is being evaluated for 1,000 samples until being replaced by a new set of 100 challenges.

Figure 1a shows a snippet of our experiment. The blue line is the core temperature gathered from the built-in SYSMON hard macro. The red line is the average inconsistency of a set of 100 test CRPs. After the 50th sample, we intentionally increase the core temperature by 5 °C using excessive switching circuits [8]. An intuitive conclusion can be drawn from the figure that the pattern of both core temperature (blue line) and average CRP inconsistency (red line) are correlated. Analysis on the entire CRP test set shows that the CRP inconsistency and collected core temperature are correlated with a correlation coefficient of 0.7461. When the core temperature increases, a rise in CRP inconsistency occurs. The inconsistency falls back as the core temperature recovers. The 5 °C increment in the core temperature leads to approximately 13.5% increment in CRP inconsistency.

However, the CRP test set is chosen at random in our experiment, a large number of stable and ultra unstable CRPs were included. Stable CRPs are always consistent regardless of the increment in the core temperature, thus greatly reduces the environmental sensitivity of the APUF. Ultra unstable CRPs on the other hand always provide near 50% inconsistency, which add additional noise to our evaluation.

2) Core Voltage vs. CRP inconsistency: Core supply voltage inside a reconfigurable device is not static. In most cases, core supply voltage VCCINT varies in a tolerable range. Many events could lead to abnormal variations in VCCINT: simple power analysis(SPA) could lead to an abnormal decrease in the VCCINT since power measurement tool introduces additional resistance to the power rail; Electrostatic discharge(ESD) could lead to abnormal increment on VCCINT etc. APUF serves as a good VCCINT detector because a slight variation in supply power results in the remapping of some CRP in APUF, thus by actively monitoring the mapping of APUF CRP inconsistency, variations in internal supply power can be detected through calculation on CRP inconsistency.

We produce a SPA scenario that results in a sudden decrease in VCCINT. The experiment Virtex-5 board uses a power module(TI TPS54620) to provide power to FPGA core with the internal supply voltage(VCCINT). The power module is essentially a buck converter that generates output voltage 0.8V $<$ V_{out} $<$ 15V. Figure 2 shows a simplified diagram of the module modified by us. We produce a SPA attack by adding an additional switch resistor to modify VCCINT directly through the power module. VCCINT can be instantly decreased by opening the switch SW to add a resistor R_{SW} into the circuit. The output voltage of the power module before opening the switch SW can be calculated using equation 3. When mimicking the probe insertion, we open the switch SW and the output voltage VCCINT can thus be calculated using equation 4.

$$\text{VCCINT} = \frac{R_1 \cdot V_{ref}}{R_2} + V_{ref} \tag{3}$$

$$\text{VCCINT} = \frac{(R_{sw} + R_1) \cdot V_{ref}}{R_2} + V_{ref} \tag{4}$$

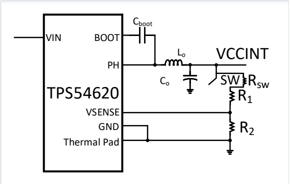


Fig. 2: VCCINT Power Module, when switch SW is open resistor R_{SW} is inserted into the power rail.

Xilinx Virtex-5 allows VCCINT to vary as much as 5% [9]. We use three configurations of R_{SW}. The replacement of resistors is capable of changing the VCCINT by 2.08%, 4.26%, and 5.37%. We apply the same evaluation as the previous experiment on the same 1,000,000 randomly selected test challenges and observe the relationship between CRP inconsistency and VCCINT. A visualized result for CRP inconsistency for a sudden change of 5.37% in VCCINT is shown in figure 1b. After the 150th voltage sensor sample, the switch is opened so that VCCINT decreased from 0.986V to 0.935V. We observe that the average CRP inconsistency increased from 23% to roughly 41% inconsistency, which later drops back to 22% as we closed the switch and restored VCCINT back to 0.986V.

B. Environmentally Sensitive CRP

The above evaluation above shows that CRP inconsistency in an APUF is correlated with core temperature and supply voltage. However, the correlation is not strong enough to meet our application needs. After review the whole CRP set, we conclude that not all CRPs are equally sensitive to the environment. Based on the notation given in equation 1 and 2, we define stable, ultra unstable and ESCRPs as below.

A stable challenge produces consistent response regardless of operating environment. A stable CRP appears if one or more segments dominate over all remaining segments so that the final delay difference is sufficiently large enough to overcome delay variations. Stable challenges can be defined as:

$$|\sum_{i}^{n}(\Delta d_i^c)| > \sum_{i}^{n}|\Delta d_i^e| \text{ for all possible } \Delta d_e \tag{5}$$

so that the APUF generate the same response regardless of environmental variation. Our APUF implementation shows that roughly 30% of CRPs are stable.

An ultra unstable challenge on the other hand creates two paths with near zero delay difference. This can be represented by:

$$\sum_{i=1}^{n}\Delta d_i^c + \Delta d_i^e \approx 0 \text{ for } \Delta d_e \in [-e, e] \tag{6}$$

With the given challenge, there is a near 50% probability that the corresponding response would be a 1 or a 0. Our APUF implementation shows that roughly 5-10% of CRPs are ultra unstable.

For the purpose of being an indicator of environmental changes, we are primarily interested in searching for environmentally sensitive challenges(ESCs) that neither produces a stable nor completely unpredictable response. An ideal ESC should at least fulfill the following two conditions:

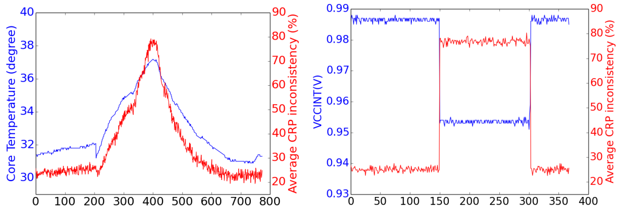
1)
$$|\sum_{i}^{n}(\Delta d_i^c)| < \sum_{i}^{n}|\Delta d_i^e| \text{ for } \Delta d_e \in [-e, e] \tag{7}$$
when so that no dominating segment group exists in the APUF, and environmental variations could possibly invert the response.

2)
$$\sum_{i=1}^{n}\Delta d_i^c + \Delta d_i^e \approx 0 \text{ for } \Delta d_e \in [-e, e] \tag{8}$$
At a given environment the response is stable in a normal operation environment.

In additional to the previous two requirements, an ideal ESC should be able to obtain an inverted response when the environment varies to an abnormal range:

3.

$$sign(\sum_{i=1}^{n}\Delta d_i^c + \Delta d_i^e) = -sign(\sum_{i=1}^{n}\Delta d_i^c + \Delta d_i^{e\prime}) \tag{9}$$
$$\text{for } \Delta d_e \in [-e, e] \text{ and } \Delta d_i^{e\prime} > e \text{ or } \Delta d_i^{e\prime} < -e.$$



(a) Core temperature(°C) vs. APUF CRP inconsistency(%) for a set of 100 ESCs.

(b) VCCINT(Volt) vs. APUF CRP inconsistency(%) for a set of 100 ESCs.

Fig. 3: Core environment change leads to variations in inconsistencies of ESCRP. The y-axis on the left is core temperature corresponding to the blue line, the y-axis on the right is the average CRP inconsistency corresponding to the red line.

In our experiment, we defined normal operating environment as: 1. core temperature $T = 30.5 \pm 5\,°\text{C}$, 2. VCCINT = $1 \pm 0.5V$. From the 1,000,000 test challenge set, we observe that no challenge fulfills all three requirements for ESCs. Non-uniform physical properties over the chip makes qualifying both equation 8 and 9 extremely difficult. We relaxed the requirement on these two conditions by 30% meaning allowing at most 30% violations in both equations. Thus we are able to collect 10,000 qualified CRPs out of a 1,000,000 random challenge set. We repeated the evaluation with only these challenges again and a snippet of the result can be seen in figure 3.

Figure 3a shows the CRP inconsistency of the ESCRPs (red line) dramatically increases when the core temperature (blue line) is intentionally increased through circuit switching. The correlation coefficient between CRP inconsistency and core temperature increased from 0.7461 to 0.8722, much higher comparing to using a random challenge set. The average CRP inconsistency increased from 23% to roughly 79.8% inconsistency when we increase the core temperature to 37.5 °C, which later drops back to 22% as switching circuits cool down. A similar observation can be made as we close the switch and change VCCINT instantly. The average inconsistency of the selected CRPs instantly changed to over 80%. Comparing to randomly selected CRPs, these sensitive CRPs response much faster and dramatic due to the fact that stable CRPs have been eliminated. We conclude that ESCRPs are much efficient and environmentally sensitive when using APUF as environmental variation indicator.

C. ESC Set Generation

APUFs serve as a good environmental variation detector when ESCs are applied. Since ESCs are different from APUF to APUF and there is no general pattern in them, acquiring a large set of ESCs at run time is no trivial task.

We present an efficient, two-step method to collect such a large set of ESCs. We first do a random search to collect a single ESC seed using evolution strategies (described in Algorithm 1). Evolution strategies method is inspired by the evolutionary adaptation of a population of individuals to certain environmental conditions. When applying evolution strategies on ESC generation, one individual in the population is a specific challenge vector and mutation on a individual is defined as randomly flipping multiple bits. The environmental fitness of the individual is determined by the consistency of the corresponding PUF response under a small environmental variation benchmark where environment factors vary beyond tolerable range using switching circuit/power rail resistor. The tolerable range is defined by the user based on the nature of the application. The pseudocode of ESC exploration is described in Algorithm 1)

Algorithm 1 Explore ESC seed

```
Require: (1) Number of candidates in the parent generation $\mu$.
    (2) Number of candidate solutions generated from the parent
    generation $\lambda$. (3) Expected ESC set size $\kappa$ (4) A fitness
    evaluation function EvaluatePopulation(). (5) A mutation
    rate $\tau$. (6) Maximum number of generations $M$.
Ensure: A set of ESCs $S_{best}$.
    Population $\leftarrow$ InitializePopulation($\mu$).
    EvaluatePopulation(Population)
    $S_{best}$ $\leftarrow$ GetBest(Population, $\kappa$)
    while $i < M$ do
        Children $\leftarrow \emptyset$
        for j = 0 to $\lambda$ do
            $Parent_j$ = GetParent(Population, j)
            $S_j$ $\leftarrow$ Mutate($Parent_j$, $\tau$)
            Children $\leftarrow$ $S_j$
        end for
        EvaluatePopulation(Children)
        $S_{best}$ $\leftarrow$ GetBest(Population+$S_{best}$, $\kappa$)
        Population $\leftarrow$ SelectBest(Population, Children, $\mu$)
        $i = i + 1$.
    end while
    Return($S_{best}$)
```

We observe that when inverting a small number of bits in an ESC seed, the new challenge is most likely to stay environmentally sensitive due to the fact that nearby transistor process variations are somewhat correlated. We thus generate more ESCs by randomly inverting a small number of bits in the ESC seed (described in Algorithm 2).

Algorithm 2 Generate ESC set

```
    Create empty challenge set $ESC$, define the number of the
    size of the ESC set $m$, and a growth index $i$. An ESC seed
    $esc$ is taken as an input
    Count = 0
    while Count ¡ $m$ do
        for $k$ in range($i$) do
            Randomly flips $k$ bits in $esc$ to obtain $c^k_{Count}$
            Add $c^k_{Count}$ to $ESC$
        end for
        Count += 1
    end while
```

V. The Detection Framework

A. System Design

Figure 4 shows a high level implementation of APUF-based abnormality detection framework. The detection framework consists of three major modules.

The challenge querier is used to generate challenges, and feed them into the APUF at run-time. Before the actual run-time detection, a calibration process generates an ESC seed off-line using algorithm 1. The challenge querier generates a set of ESCs using algorithm 2 at run-time. A random number generator embedded inside of the challenge querier decides at each clock cycle which bits should be inverted.

The APUF process the same challenge multiple times and send all generated responses to the response verifier module. The APUF itself resides immediately next to the logic being monitored to achieve the best accuracy.

The response verifier calculates the CRP inconsistency by counting the frequency of response inversion, and raise an alarm if the inconsistency exceeds a user defined threshold. Since all ESCs are correlated, the 0 and 1 distribution are also very similar, thus a single counter is enough for counting and recording bit inversions. The alarm threshold is recommended to be set to the selected ESC's fulfillment rate of equation 8 and 9, however it is adjustable as the definition of the normal operational environment may be subject to change based on user requirement.

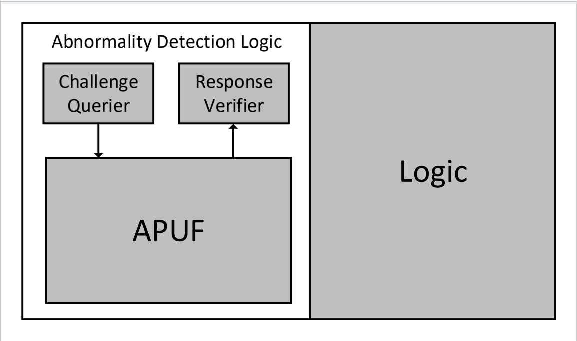


Fig. 4: APUF-based abnormality detection framework.

B. Experimental Results

In this section we carefully evaluate our abnormality detection framework.

Our experiments are conducted on two Genesys boards. The Genesys board has a Virtex-5 LX50 FPGA. The system is tested with an implementation of an RSA-1024 and an AES-128 crypto-system [10]. A mixture of 1,000 abnormal core temperature and VCCINT was applied during a 10-hour long operation. In our setting, an abnormality in core temperature is defined as a variation of over 5 °C and an abnormality in VCCINT is defined as a variation of over 5%. We generated three distinctive challenge sets, respectively 1,000 random challenges(RC), a single ESC seed, and 1,000 ESCs derived from the ESC seed. The ESC seed is generated based on our

	Virtex-5 SYSMON[6]	Our design on Virtex-5	RO-based monitor [7]	Our design on Spartan-6
Sample rate	200kHz	200kHz	8MHz	8MHz
LUTs	251	92	3300	49
Flip-flops	139	69	unknown	57
Slices	63	36	825	22
Power(mW)	653	569	1953(estimated)	29

TABLE I: FPGA resource and power characteristics: Virtex-5 SYSMON vs. our design on Virtex-5 vs. RO-based on-chip power monitor vs. our design on Spartan-6. ADC and analog sensor area and power are not included in calculation.

requirement on core temperature and VCCINT, which in our case is our definition of the normal range of temperature and VCCINT defined above. The alarm threshold is set to raise an alarm when CRP inconsistency reaches 40% for 1,000 random challenges, 70% for both single ESC seed and 1,000 ESCs case. A UART core is used to monitor the real-time CRP inconsistency and communicates with RSA/AES cores. The results are displayed in Table II.

Circuit	1,000 RC	Single ESC	1,000 ESC
AES-128	69.4/0.7	98.1/5.9	100/2.1
RSA-512	78.4/3.0	100/3.9	100/1.1

TABLE II: Abnormality detection: true positive rate (%)/false positive rate(%) using 1,000 random challenges vs. single ESC vs. 1,000 ESCs as challenge set per sample. APUFs are placed immediately besides the monitored circuit. A mixture of 1,000 abnormal temperature and VCCINT were applied to the protected circuit.

We observe that 1,000 ESC challenge set provides the best detection rate while random CRPs provides the worst. ESCs benefit from the elimination of stable and ultra unstable challenges. With a good ESC seed, we are capable of achieving a true positive rate of 100% with 1,000 generated ESCs at runtime, while the false positive rate is as low as 1.1%.

C. Area and Power

We compare our design's area and power overhead with both Xilinx System Monitor [6] and ring oscillator-based power monitor[7]. Since Xilinx System Monitor only supports Virtex family product and Le Masle's ring oscillator-based power monitor was originally implemented on Spartan-6 LX45, we implement our design on both platforms for comparison purposes. The sample rate for Virtex-5 implementations is fixed to 200kHz while the implantation on Spartan-6 devices has a sample rate of 8MHz. The FPGA resource and power characteristics for Xilinx System Monitor, ring oscillator-based power monitor and our design are shown in table I.

Our design uses 63% less FPGA area comparing to Xilinx System Monitor logics while achieving 13% of power savings. To be noted that we do not include area and power of both analog sensor and ADC in the SYSMON hard macro in our calculation due to lack of information. The actual saving is expected to be more. Comparing to Le Masle's ring oscillator-based power monitor, our design is 98.4% smaller. Since APUF does not impose high switching activities like ring oscillators, our design consumes only 1.5% of the power.

VI. Conclusion

In conclusion, we discovered that APUF CRP inconsistency is highly correlated with the core temperature and voltage on reconfigurable platforms. Based on the observation we designed a framework to utilize the environmental sensitivity of APUFs to detect suspicious operational environment variations. We propose to apply only ESCs on a given APUF to efficiently and accurately detect environmental abnormalities. When integrated with an AES-128 and an RSA-1024 implementation, our framework is capable of detecting 100% of applied abnormalities with false positive rate as low as 1.1%. Our design provides competitive detection rate with both sensor-based Xilinx System Monitor and Le Masle's ring oscillator-based power monitor while using much less hardware and power.

VII. Acknowledgement

This work was supported in part by the NSF under award CNS-1059435, and award CNS-1513306.

References

[1] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical one-way functions," *Science*, vol. 297, no. 5589, pp. 2026–2030, 2002.

[2] B. Gassend, D. Clarke, M. Van Dijk, and S. Devadas, "Silicon physical random functions," in *Proceedings of the 9th ACM conference on Computer and communications security*. ACM, 2002, pp. 148–160.

[3] T. Xu and M. Potkonjak, "Robust and flexible FPGA-based digital PUF," in *Field Programmable Logic and Applications (FPL), 2014 24th International Conference on*. IEEE, 2014, pp. 1–6.

[4] H. Gu, T. Xu, and M. Potkonjak, "An Energy-Efficient PUF Design: Computing While Racing," in *Proceedings of the 2016 International Symposium on Low Power Electronics and Design*. ACM, 2016, pp. 142–147.

[5] T. Xu, H. Gu, and M. Potkonjak, "An ultra-low energy PUF matching security platform using programmable delay lines," in *Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC), 2016 11th International Symposium on*. IEEE, 2016, pp. 1–8.

[6] Xilinx, *Virtex-5 FPGA System Monitor User Guide*, Xilinx.

[7] A. Le Masle and W. Luk, "Detecting power attacks on reconfigurable hardware," in *22nd International Conference on Field Programmable Logic and Applications (FPL)*. IEEE, 2012, pp. 14–19.

[8] M. Happe, H. Hangmann, A. Agne, and C. Plessl, "Eight ways to put your FPGA on fireA systematic study of heat generators," in *Reconfigurable Computing and FPGAs (ReConFig), 2012 International Conference on*. IEEE, 2012, pp. 1–6.

[9] Xilinx, *Virtex-5 FPGA Data Sheet:DC and Switching Characteristics*, Xilinx.

[10] "Cryptographic Hardware Project - cores," http://www.aoki.ecei.tohoku.ac.jp/crypto/web/cores.html, accessed: 2017-01-12.