Zhang JL, Qu G, Lv YQ *et al.* A survey on silicon PUFs and recent advances in ring oscillator PUFs. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 29(4): 664–678 July 2014. DOI 10.1007/s11390-014-1458-1

A Survey on Silicon PUFs and Recent Advances in Ring Oscillator PUFs

Ji-Liang Zhang[1,2] (张吉良), *Student Member*, *CCF*, *ACM*, *IEEE*, Gang Qu[1,*] (屈　钢), *Senior Member*, *IEEE*
Yong-Qiang Lv[3,4] (吕勇强), and Qiang Zhou[3,4] (周　强), *Senior Member*, *CCF*, *IEEE*

[1] *Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742, U.S.A.*

[2] *College of Information Science and Engineering, Hunan University, Changsha 410082, China*

[3] *Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China*

[4] *Research Institute of Information Technology, Tsinghua University, Beijing 100084, China*

E-mail: hnu.zjl@gmail.com; gangqu@umd.edu; {luyq, zhouqiang}@mail.tsinghua.edu.cn

Received March 26, 2014; revised May 5, 2014.

Abstract Silicon physical unclonable function (PUF) is a popular hardware security primitive that exploits the intrinsic variation of IC manufacturing process to generate chip-unique information for various security related applications. For example, the PUF information can be used as a chip identifier, a secret key, the seed for a random number generator, or the response to a given challenge. Due to the unpredictability and irreplicability of IC manufacturing variation, silicon PUF has emerged as a promising hardware security primitive and gained a lot of attention over the past few years. In this article, we first give a survey on the current state-of-the-art of silicon PUFs, then analyze known attacks to PUFs and the countermeasures. After that we discuss PUF-based applications, highlight some recent research advances in ring oscillator PUFs, and conclude with some challenges and opportunities in PUF research and applications.

Keywords physical unclonable function, hardware security, trusted IC, VLSI, FPGA

1 Introduction

With the increasing demands of security, privacy protection, and trustworthy computing, device authentication and cryptographic key storage become two of the most challenging design concerns, particularly for systems such as smart cards, sensors, and smart phones where the lack of persistent power limits the duration of countermeasure enforcement and the choices of places for key storage. Silicon physical unclonable function (PUF) is a promising solution for both challenges. "PUF is a physical entity that is embodied in a physical structure and is easy to evaluate but hard to predict. Further, an individual PUF device must be easy to make but practically impossible to duplicate, even given the exact manufacturing process that produced it"①. It is well know that the threshold voltage and gate oxide thickness on each logical gate may not be the same, which means that chips built under the same conditions will be different in terms of performance metrics such as delay and power because of this fabrication variation. As the technology keeps scaling down, such differences become more and more significant, affecting not only chip performance, but sometimes also circuit functionality. Therefore, in the past couple of decades, there have been many efforts in attempt to reduce or control manufacturing variation. Starting from 2001, with the introduction of physical one-way functions[1] and the silicon physical random functions[2-3], researchers have discovered methods to put such variation for good uses such as device authentication, cryptographic key storage, and various security applications. Most PUFs provide a unique device-dependent mapping from a set of challenges to a set of responses (challenge response pair, or CRP) based on the unclonable properties of the underlying physical device. They should satisfy the following three properties[4]:

• *Persistent and Unpredictable.* The response R to a challenge C is random and unpredictable, but it should

Survey

This paper is supported in part by the National Natural Science Foundation of China under Grant No. 61228204, the scholarship from China Scholarship Council under Grant No. 201306130042, and the Ph.D. Candidates' Innovative Research Project of Hunan Province of China under Grant No. CX2012B142.

*Corresponding Author

① Physical unclonable function. http://en.wikipedia.org/wiki/Physical_unclonable_function, May 2014.

remain the same for the same challenge over multiple observations.

• *Unclonable.* It is impossible to obtain R from C without the physical presence of the PUF. In other words, given a PUF, it is infeasible for an adversary to build another PUF that provides the same responses to every possible challenge. This is assumed to be true due to the uncontrollable technology variations.

• *Tamper Evident.* Invasive attacks to PUFs will destroy the PUFs and thus can be detected easily.

In this article, we survey the current state-of-the-art of silicon PUFs, highlight some recent research advances in ring oscillator PUFs, and discuss the challenges and opportunities in PUF research and applications. The rest of this paper is organized as follows. The detailed PUF taxonomy is presented in Section 2. The PUF evaluation criteria are described in Section 3. The known attacks to PUFs and the countermeasures are elaborated in Section 4. The basic PUF-based applications are then given in Section 5. Section 6 introduces recent research advances in the design and implementation of ring oscillator (RO) PUF in detail. We introduce and discuss some recent challenges and new opportunities in Section 7. Finally, we conclude the paper in Section 8.

2 PUF Taxonomy

There has been more than a decade of intensive study on PUFs since the concept was first introduced in [1] by Pappu *et al.* Among the many PUFs that have been proposed, silicon PUFs are of the most interest in terms of fabrication cost and readiness to be integrated to computing and communication devices. There are three major silicon PUFs defined by the physical features that generate them. They are analog electronic PUFs, memory-based PUFs, and delay-based PUFs. The focus of this article will be on one representative delay-based PUF built with ring oscillators (RO). In this section, we give a brief introduction to silicon (non-silicon) PUFs and strong (weak) PUFs as outlined in Table 1.

Table 1. Detailed PUF Taxonomy

Non-silicon PUF	Optical PUF, paper PUF, CD PUF, acoustical PUF, magnetic PUF, RF-DNA, etc.
Silicon PUF	Analog electronic PUF (ICID, power grid PUF, coating PUF, LC PUF, etc.), memory-based PUF (SRAM PUF, butterfly PUF, latch PUF, flip-flops PUF, bistable ring PUF, MECCA PUF, etc.), delay-based PUF (arbiter PUF, ring oscillator PUF, glitch PUF, IP-PUF, HELP, etc.)
Strong PUF	Arbiter PUF, optical PUF, lightweight secure PUF, bistable ring PUF, IP-PUF, etc.
Weak PUF	ICID, coating PUF, ring oscillator PUF, SRAM PUF, butterfly PUF, latch PUF, flip-flop PUF, etc.

2.1 Non-Silicon PUFs

Non-Silicon PUFs mainly includes the optical PUF[1], acoustical PUF[5], paper PUF[6-7], CD PUF[8], magnetic PUF[9], RF-DNA[10] and phosphor PUF[11-12].

Optical PUF. Pappu *et al.*[1] introduced the idea of Physical One-Way Function (POWF). They used a bubble-filled transparent epoxy wafer with three-dimensional (3D) micro-structure and shined a laser beam through it resulting in a two-dimensional (2D) speckle pattern. The speckle pattern is then filtered by a Gabor transform to produce a fixed-length bit key. POWF is a kind of optical PUF. The challenge of the optical PUF is the precise angles of laser beam and the response is a fixed-length key. Tuyls *et al.*[13-14] conducted information-theoretic analysis of the optical PUFs. The authors also introduced "slow PUF" as a way to make brute force attacks difficult. The word "slow" refers to the long measurement time for the attackers. Ignatenko *et al.*[15] present a method for estimating the secrecy rate of PUFs using a universal source coding algorithm called Context-Tree Weighting method. Based on measurement results on optical PUFs, a secrecy rate of 0.3 bit/location is estimated.

Acoustical PUF. Vrijaldenhoven introduced acoustical PUF[5] by exploiting the manufacturing variation between small plates of some kind of material (e.g., glass). For an acoustical PUF, an electrical signal (oscillating voltage) is transformed to a mechanical vibration through a transducer. This vibration propagates as a sound wave through the token and scatters on the randomly distributed inhomogeneities. The wave arrives at another transducer which converts the wave to an electrical signal again. The signals that result from this scattering will be unique for each acoustical PUF.

Paper PUF. Since the intrinsic roughness presents on all non-reflective surfaces can be used as a source of physical randomness which can potentially provide strong, in-built, hidden security for a wide range of paper, plastic or cardboard objects, Buchanan *et al.*[6] proposed to shine a focused laser beam through document at a specific angle which results in reflected intensity that can be used as fingerprint to prevent document and branded-product against fraud. Verifying fingerprints requires high-powered laser microscope, which may be prohibitively expensive for most users.

RF-DNA. DeJean *et al.*[10] proposed an RF-DNA technology which generates unclonable physical fingerprints based on the subtleties of the interaction of devices when subjected to an electromagnetic wave.

These fingerprints can help to produce a cryptographic certificate of authenticity which, when associated with a high value good, may be used to verify the authenticity of the good and to distinguish it from any counterfeiting goods.

Phosphor PUF. Chong *et al.*[11-12] used a phosphor PUF to construct anti-counterfeiting systems by using a random pattern formed by any scattering phosphor particles as the physical identifier.

Magnetic PUF. Indeck and Muller[9] proposed magnetic PUF which employs the inherent uniqueness of the particle patterns in magnetic media to generate unique fingerprint.

2.2 Silicon PUFs

Silicon PUFs utilize the uncontrollable manufacturing variations to generate a unique signature for each IC. According to the different source of variation, silicon PUFs can be categorized as analog electronic PUFs, memory-based PUFs, and delay-based PUFs.

2.2.1 Analog Electronic PUFs

Analog electronic PUFs mainly include ICID[16], the coating PUF[17], silicon nanokey[18], LC-PUF[19] and power grid PUFs[20].

Lofstrom *et al.*[16] proposed to exploit fluctuations in drain currents to create IC identification (ICID) which is a unique identification for each manufactured IC. However, this ICID method has limited IDs (actually is a weak PUF). Hence, it is not secure since adversaries can exhaustively readout the ID and "clone" an IC. Tuyls *et al.*[17] introduced coating PUFs that are resistant against invasive attacks since this protective coating is opaque and chemically inert. The protective coating consists of a matrix material doped with random dielectric particles covering an IC. The top metal layer of the IC contains an array of sensors that are used to measure the local capacitance values of the coating. A challenge corresponds to a voltage of a certain frequency and amplitude applied to the sensors at a certain point of the sensor array. The response is the measured capacitance value which is converted into a bit string as a key or an ID. Škorić *et al.*[21] introduced a physical model of coating PUFs by simplifying the capacitance sensors to a parallel plate geometry. The entropy of the probability distribution function of the capacitance can be estimated by using the model. Puntin *et al.*[18] proposed a silicon nanokey that exploits the variability of the electrical parameters of minimum size MOS transistors, in particular of the threshold voltage, to generate a PUF to be used in a challenge-response authentication scheme.

2.2.2 Memory-Based PUFs

Static RAM PUF (SRAM PUF)[22-23] and the butterfly PUF[24] are two representative memory-based PUFs.

The SRAM PUF[23] consists of a large number of memory units. As shown in Fig.1, a memory unit is formed of two cross-coupled inverters with two stable states that are normally represented by 0 and 1. Any minor voltage difference that shows up on the transistors due to intrinsic parameter variations will tend toward a 0 or a 1 caused by the amplifying effect of each inverter acting on the output of the other inverter[23]. The challenge refers to a subset of memory units that is read after being powered up; the response means the state of the subset as a result of being powered up. Because not all FPGAs support memories that do not need initialization, the SRAM PUF is not suitable for all types of FPGA. In order to resolve the problem, Guajardo *et al.*[24] proposed an improved SRAM PUF[31], named butterfly PUF. While units in SRAM PUF are based on a cross-coupled inverter, the butterfly PUF uses an unstable cross-coupling circuit, replacing an inverter with a latch or flip-flop, as shown in Fig.2. The latch of butterfly PUF, as a circuit for storing information, can be emptied (the output is 0) or be reset

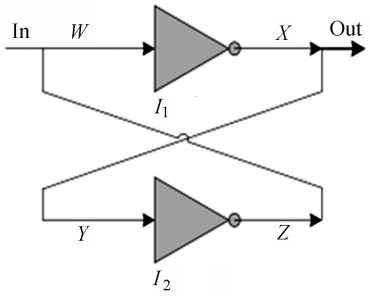


Fig.1. Two cross-coupled inverters[31].

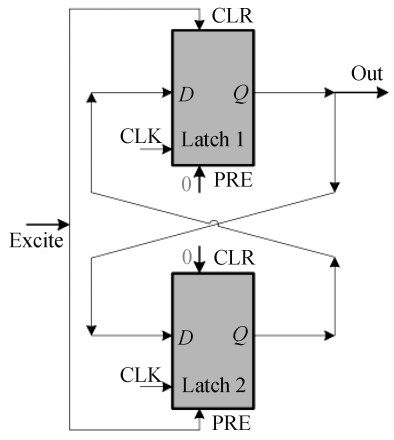


Fig.2. Butterfly PUF: cross-coupled latches[31].

(the output is 1), bringing the advantage that they do not require measurement by being powered up. Therefore, the butterfly PUF is applicable to all types of FPGA. Besides, the SRAM PUF is vulnerable to the attack of exhaustive readout. To address this issue, Krishna *et al.* proposed a memory-based PUF, MECCA PUF[25], where the write pulse duration is used as challenge, and write failures in memory cells would generate PUF response. However, MECCA PUF is only suitable for embedded memory of specific structures[26]. Holcomb *et al.*[22] proposed a similar idea to SRAM PUFs[23], but it focuses on using the initialization SRAM state in RFID tags to create a physical fingerprint. The advantage is that it can be implemented using the existing SRAM memory cells of the RFID chip without the need for additional hardware. Maes *et al.*[27] proposed to build a flip-flop PUF by capturing random start-up values of flip-flops present in the reconfigurable logic of a commercial FPGA. The main advantage is that the flip-flop PUF is truly intrinsic (can be used on any SRAM-based FPGA) and does not consume any resources. SRAM PUFs are also particularly resilient to temperature variations and are generally more compact than the flip-flop and butterfly PUFs[28]. However, SRAM PUFs have been reported being physically cloned recently[29].

2.2.3 *Delay-Based PUFs*

There exist many delay-based PUFs that include the arbiter PUF[30-32], ring oscillator (RO) PUF[33-35], glitch PUF[36-37], HELP[38], Intrinsic Personal PUF (IP-PUF)[39] and so on. Arbiter PUF and RO PUF are the most popular ones and we will focus on them.

• *Arbiter PUF*

Arbiter PUFs were first introduced in [30-31]. Their basic structure is shown in Fig.3. Two parallel 128-order multiplexer chains share the same input port and have their output ports connected to the input port D and the clock input port of a D flip-flop, respectively. The input port uses step input signals. The challenge input bits: $X[0] \sim X[127]$, enter the multiplexer chains through their select ports. Signal $X[i]$ determines which multiplexer in the i-th stage of the multiplexer chains the input step signals will go through. Different challenge input signals and the delay difference between the two parallel multiplexer chains determine whether the step signal will reach the flip-flop input port D or the clock input port. In the former case, a logic-1 will be latched, and a logic-0 will be latched in the latter case. This latched value can serve as a 1-bit PUF signature or response to the challenge.

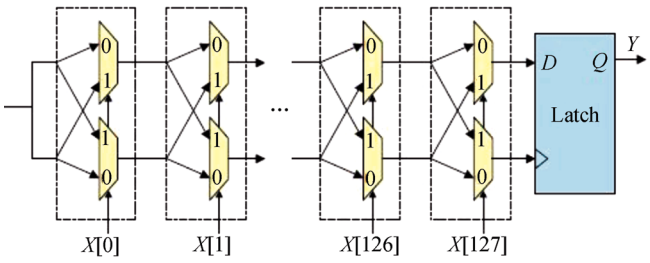


Fig.3. Structure of Arbiter PUF[33].

Since the response from a delay-based silicon PUF can be represented by a linear function of the challenge, an attacker who knows the delay of each unit in a circuit path can predict the response corresponding to a given challenge by calculating the sum of delays of all units[32]. Though it is difficult to measure the delay of each unit, the attacker can build a software model to simulate the arbiter PUF by using methods such as machine learning[40-43] and predict the response to a random challenge. Therefore, the arbiter PUF is vulnerable to model-building attacks. In order to resist the non-invasive model building attacks, Gassend *et al.* proposed to add a feed-forward arbiter[44] scheme to make the modeling task much harder. However, Majzoobi *et al.*[45] proposed four test methods for evaluating security of PUFs, including predictability, collision, sensitivity, and reverse engineering. The test results show that popular linear arbiter PUFs and feed-forward arbiter PUFs can be reverse engineered and emulated. Therefore, they proposed a lightweight secure arbiter PUF[46] that uses multiple delay lines for creation of PUF responses and uses judicious combination of challenge input bits to drastically reduce controllability. They subjected the outputs from multiple delay lines to a scrambling lossy transformation to create modular, easy to parameterize, secure, and reliable PUF structures.

In addition, the arbiter PUF requires the routing of the two multiplexer chains to be completely symmetric, which would otherwise dominate the effect of manufacturing variations. Hence, the arbiter PUF is difficult to be implemented on FPGAs. The uniqueness of the arbiter PUF implemented on an FPGA is only 1.05%[47]. In order to balance FPGA routing asymmetries, Majzoobi *et al.*[48] proposed to construct an FPGA PUF using programmable delay line (PDL) implemented by lookup table. Ozturk *et al.*[49] also proposed to use tristate buffers to create the delay chain. The tristate buffer PUF is extremely similar to the arbiter PUF, except that tristate buffers are used in place of multiplexors in the delay chain. The advantage is that the tristate buffer PUF circuit consumes less power and requires less area than the arbiter PUF.

• *RO PUF*

In 2007, Suh *et al.* proposed the ring oscillator (RO) PUF[33] which is based on the delay difference among ROs to generate random bit strings. An RO is a simple circuit of a set of inverters connected in a loop, as shown in Fig.4, that oscillates with a particular frequency. The frequency depends on the delay of each inverter and the wires, which cannot be predicted due to manufacturing process and other uncertain factors. The simplest form of PUF generates the output logic-0 or logic-1 by comparing the frequencies of a pair of oscillator circuits (see Fig.4). More bits can be generated in the same way with multiple pairs of ROs. RO PUFs do not require high symmetry and thereby is easy to be implemented on FPGAs. However, an RO PUF consumes more resources than an arbiter PUF, while produces a limited number of challenge-response pairs, and needs hard macros to fix the routing[50]. Since RO PUF and arbiter PUF are not specially designed for FPGAs, Anderson[37] thus proposed a delay-based glitch PUF specially for FPGAs. The PUF makes use of the intrinsic structure of FPGA (look-up table and multiplexer), and it can be naturally embedded into a design's HDL and does not require the use of "hard macros" with fixed routing. However, the glitch PUF cannot be directly deployed on the new generation FPGAs, and therefore it has the scalability issue. In order to resolve the issue, Zhang *et al.*[50] designed and implemented a scalable glitch PUF on 28 nm FPGAs with relative higher hardware overhead. We will elaborate more advanced design of RO PUF later in this article.

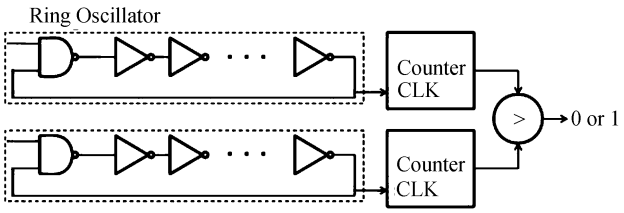


Fig.4. Basic structure of RO PUF[37].

2.3 Strong PUFs and Weak PUFs

From the above discussion, PUFs can be divided into strong PUFs and weak PUFs[40]. The strong PUF includes the optical PUF, arbiter PUF, lightweight secure PUF, etc. The weak PUF mainly includes the memory-based PUF, RO PUF and glitch PUF. The security of strong PUFs is based on their high entropy content providing a huge number of unique challenge-response pairs (CRPs), which can be used in authentication protocols. On the other hand, weak PUFs exhibit only a small number of CRPs to be applied. Although they are not applicable to authentication protocols, the corresponding responses of weak PUFs can be used as a device unique key or seed for conventional encryption systems, while maintaining the advantages of physical unclonability. In order to enable the extraction of cryptographic keys from PUFs, the fuzzy extractor[51] is necessary.

3 PUF Evaluation Criteria

The quality and hence usability of a PUF is affected by the following four main factors.

• *Hardware Efficiency in Implementation.* The hardware overhead may make the silicon PUF impractical when its implementation requires a large amount of additional circuitry to be implemented on ASIC or FPGA. Designing PUF with hardware efficiency is critical for its usability.

• *Reliability Subject to Operating Environment Variations.* Reliability measures the stability of PUF responses in different environments. Ideally, PUF signatures should remain the same under the same challenge no matter when, where and how many times the challenge is presented. But in reality, a variety of environmental conditions, such as temperature, voltage, and the aging of devices, may lead to changes in the circuit delay and other physical characteristics that PUFs are defined on. This will cause PUF signatures to vary. The difference between any two signatures generated by a PUF under the same challenge in repeated experiments should be slight. The following formula[50] can be used to evaluate the reliability of PUFs:

$$r = \frac{1}{x}\sum_{y=1}^{x}\frac{HD(R_i, R_{i,y})}{n} \times 100\%, \tag{1}$$

where x stands for the number of samples, n is the number of bits of a signature generated by the PUF, and $HD(R_i, R_{i,y})$ is the Hamming distance between the response R_i and the y-th sampling $R_{i,y}$.

• *Security Against Generic Attacks.* The uniqueness, uniformity, and randomness of PUF responses are related to the security of a PUF. The uniqueness shows how unique a PUF response can be, which determines the quality of the PUF. It is not acceptable if different PUFs produce the same or very similar responses when fed with the same challenge. Hamming distances[50] (HDs) are used to evaluate PUF response's uniqueness. For a pair of PUFs: P_i and P_j $(i \neq j)$ that both generate n-bit signatures, their average Hamming distance will be calculated as follows.

$$u = \frac{2}{k(k-1)}\sum_{i=1}^{k-1}\sum_{j=i+1}^{k}\frac{HD(P_i, P_j)}{n} \times 100\%. \tag{2}$$

Uniformity is used to describe the distribution of "0" and "1" in a PUF signature. The uniformity of PUF signatures generated by a PUF is associated with the security of the PUF. In extreme cases, if all bits of a response generated by a PUF are "0" or "1", then all the responses generated by the PUF will be identical and cannot be unique. As long as the distribution of "0" and "1" in a PUF signature is uneven, the security of the PUF will be affected to some extent[50]. The ideal uniformity is achieved when "0" and "1" occurs with equal probability for each bit in a PUF signature. Randomness is also used to evaluate the security of PUF. Usually, the randomness of PUF responses can be tested by the NIST statistical test suite.

• *Security Against Known Attacks.* The security is to prevent an adversary from acquiring the PUF secret or CRPs by side channel analysis[41,52-56], machine learning techniques[40,42,57] and the physical cloning attacks[29]. Due to PUF information's role in security-related applications, ensuring that the PUF CRPs are secure and unclonable is vital for PUF's usability. We will give a detailed security analysis of PUF against known attacks in the next section.

4 Known Attacks to PUFs and the Countermeasures

In recently years, the basic assumptions of PUFs such as unpredictable, unclonable, and tamper evident have been questioned and thus the security issue of PUFs has attracted great amount of attentions. Several attacks on PUF properties have been reported including modeling attacks, side channel attacks, and physical cloning attacks. Some of these were specially developed to attack a specific PUF and cannot be applied to every PUF. However, if there are no effective solutions to solve the challenges of hardware overhead, security, and reliability, PUF is likely to become a failed hardware security primitive.

1) *Modeling Attacks.* Machine-learning (ML) based modeling attacks are one of the best-known PUF attacks. They are exclusively applicable to strong PUFs. Strong PUFs has a publicly accessible CRP interface, which allows the simple collection of the large number of CRPs that are required in this attack type during their learning phase. Machine learning techniques have been reported that it can model some strong PUFs with high prediction rate. Ruhrmair *et al.*[40,42] demonstrated modeling attacks on RO PUFs, arbiter PUFs, and arbiter variants by using machine learning techniques. Experimental simulation shows that the approach can foresee PUF responses to a given challenge with prediction rates up to 99%. Mahmoud *et al.*[57] combined machine-learning based modeling techniques with side channel information leak to attack strong PUFs such as XOR arbiter PUFs and lightweight PUFs up to a size and complexity that was out of reach[40,42]. They reported successful attacks for 64-bit, 128-bit and 256-bit, and for up to nine single arbiter PUFs whose output is XORed, whereas [40, 42] attack this structure only for up to five XORs and bit-length 64. However, the modeling attacks need PUFs providing a huge number of CRPs, which makes them unapplicable to weak PUFs such as SRAM PUF[23] and similar architectures.

2) *Side Channel Attacks.* Side channel attack statistically analyzes the time, power consumption or electromagnetic emanation of the cryptographic devices to gain knowledge about integrated secrets. Karakoyunlu and Sunar[52] reported the first successful power side-channel attack on the software implementation of fuzzy extractor. This implies that software implementation of any error correction scheme is potentially vulnerable to side-channel attack and error correction can lower the security of PUF. Merli *et al.*[54] studied the side-channel analysis of silicon PUFs and their fuzzy extractors. They pointed out that the frequencies of ROs on FPGA can be measured with state-of-the-art electro-magnetic (EM) equipment and thus it becomes possible to clone the RO PUF. They also implemented attacks on the fuzzy extractor which can successfully extract the cryptographic keys generated by PUFs using fuzzy extractor. In another work[56], the same authors showed that by exploiting the chained challenges and EM emanation, it is possible to deduce the relative frequency rank of the ROs and guess correctly the PUF secret bits. More recently, the same group[58] demonstrated that it is feasible to measure the EM emission of a single tiny RO with only three inverters within a single configurable logic block on an FPGA chip. The authors also pointed out that their proposed attack can be successful because they exploited that each RO has a fixed location and a specific measurement path through a multiplexer to a counter. Hence, the authors proposed to randomize the measurement path with high overhead and interleave ROs, multiplexers, comparators, and counters on a register and lookup table level. Besides, side-channel attacking on ECCs of weak PUFs[53] is also a feasible way to break the security of PUFs. One of the countermeasures is to use code-word masking to protect PUFs error correction[59].

3) *Physical Cloning Attacks.* Recently, Helfmeier *et al.*[29] demonstrated the first successful physical cloning of an SRAM PUF based on the fact that SRAM cells emit near infrared light when it is read and the cell's power-up value can be obtained from the emitted light. Hence, SRAM PUFs are not well suited as secure PUFs. Other PUFs such as RO PUFs or arbiter PUFs have not been reported to this date.

Since silicon PUFs are based on the comparison of delay paths, they are vulnerable to model attacks, and are very sensitive to ambient environmental variations[2]. Hence, interfacing Hash functions to the PUF challenge/responses[3] were proposed to address the problems. Gassend *et al.*[60] introduced the concept of controlled PUFs which can only be accessed via an algorithm that is physically linked to the PUF in an inseparable way. This control algorithm can protect weak PUFs from external attacks since any attempt to circumvent the algorithm will lead to the destruction of the PUF. The control algorithm hardens the security by cryptographically manipulating the input and output of the PUF and preventing direct access to the PUF. The controlled PUF can be applied to some elementary applications, such as certified execution[60].

5 PUF-Based Applications

Due to the unique properties of PUFs, they have been used in a large variety of applications such as integrated circuit intellectual property (IP) protection[23-24,50,61-64], key generation[31,33,65], device authentication[18,33,66-69], digital rights management[70-71], trusted computing[72], vehicle system security[73], and so on. In this section, we first briefly survey two of PUF's most popular and well-documented applications, namely secure key storage and device authentication. Then we give a detailed introduction on how PUF can help in IP protection. We must note that PUF responses may change due to factors such as ambient temperature variation and supply voltage fluctuation since these factors may affect circuit delay in practice, and hence reliability-enhancing techniques such as string pattern matching[74], Index-Based Syndrome coding (IBS)[75] or fuzzy extractor[51] need to be used for different applications.

5.1 Secure Key Storage

With the popularization of electronic devices, we are increasingly dependent on IC to securely handle sensitive information. For example, the RFID is used as a key card to control the building, and smart cards are used to perform financial transactions. Therefore, it is important that IC can protect sensitive information. The traditional method is to store a secure key in the non-volatile memory (e.g., EEPROM) in order to use cryptographic primitives (such as digital signatures and encryption) to protect sensitive information. However, it has some obvious drawbacks. For example, the recently proposed non-invasive and invasive physical tampering techniques[76-77] (e.g., micro-probing, laser cutting, side-channel) can allow an attacker to extract the digital key stored in nonvolatile memory, and therefore compromise the cryptographic-related secure mechanisms. In order to prevent physical attacks, researchers have proposed various protection mechanisms, where PUF is a new cryptographic primitive that can produce a secure volatile key. By means of storing secrets in its unique intrinsic physical features that are randomly determined by fabrication variations, e.g., the subtle difference in the delays of two wires with equal length at the design phase, PUF achieves a higher level of protection without relying on persistent power. The validity of the claim rests on the insight that attempts in conducting invasive attacks will alter the unique intrinsic features and therefore destroy the secret hidden in the victim devices with a higher probability.

5.2 Device Authentication

Strong PUFs exhibit a huge number of challenge-response pairs (CRPs), hence they can be used for device authentication with low cost. Suh *et al.*[33] demonstrated how PUFs can be used to authenticate individual ICs without costly cryptographic primitives. As shown in Fig.5, a PUF is embedded into the device A, and CRPs are collected and stored in a secure database. As the PUF responses are unique and unpredictable for each device, we can simply compare a re-generated PUF response with the pre-stored response in the database with the same challenge. When there is a perfect match or the matched portion is more than a pre-determined threshold, the device will be considered authenticated. In order to protect against man-in-the-middle attacks, the used CRPs will be deleted from the database. Later on, many lightweight and secure PUF authentication protocols are also proposed[66-67,69]. These protocols are suited to resource-constrained environment such as embedded systems and pervasive networks.

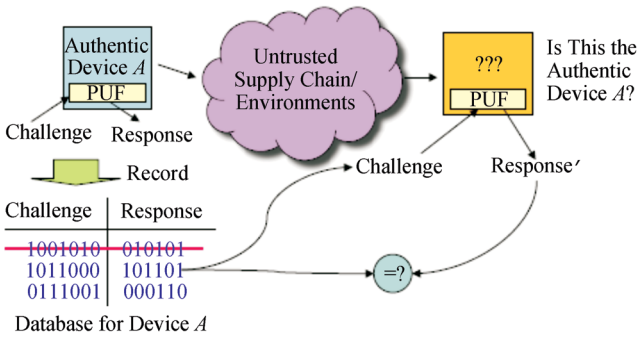


Fig.5. PUF-based low-cost authentication[33].

5.3 FPGA IP Protection

With FPGA's static and dynamic reconfigurability[78], continuous improvement in performance, and the

decrease of production cost, FPGAs have been widely used in the computing acceleration, communication, and other areas. However, hardware IP (HW-IP) which is defined as the soft-core (synthesized from HDL) hardware modules stored in the FPGA configuration bitstream[79] is vulnerable to piracy attacks. PUFs have been used as a promising hardware security primitive to resolve the issue.

Currently, several HW-IP protection techniques have been proposed to prevent piracy such as watermarking[80-82] and encryption[23,64,83-85]. However, watermarking is a passive IP protection technology, which means that it cannot actively prevent IP from being illegally duplicated, distributed, or integrated into SOC[50], and encryption-based methods are also faced with some severe issues discussed in [86]. Most recently, Zhang *et al.*[62-63,86] proposed the first nonencryption-based FPGA IP binding technique that combines the unclonable PUF signatures of hardware with the finite state machines (FSMs) of sequential circuits in FPGA designs/IP cores to actively restrict FPGA designs/IPs to running on authorized hardware platforms. The schemes can potentially address the drawbacks of the encryption-based schemes mentioned above. Meanwhile, it can provide commercially popular pay-per-device licensing mechanism[62] which provides technical support for the system developers to pay IP licensing fees only for the FPGA devices they are using. The key part of the binding method is the interaction protocol among the FPGA vendor, core vendor, system developer, and end user. The binding protocol includes four parts: 1) FPGA device enrollment; 2) HW-IP core enrollment and distributing; 3) HW-IP core licensing; 4) FPGA-based product licensing. Based on the binding protocol, the authors introduced a prototyping design and implementation of the lock mechanism proposed in the binding scheme.

6 Recent Research Advances in RO PUF

The RO PUF is a popular silicon PUF that can generate highly reliable unclonable outputs by amplifying the delay difference caused by fabrication variations through the substructure of ring oscillators. In this section, we will introduce some recent research advances in the design and implementation of RO PUF.

Fig.6 illustrates an RO PUF with N oscillators. N oscillators can produce $N \times \log(N)$-bit information entropy. The oscillators in Fig.6 must be identical, so as to ensure that the frequency differences between them are caused by the differences in random manufacturing processes. The ring oscillators are connected to the clock input ports of the two counters and obtain 1-bit PUF signature by comparing values read from the two counters within a period of time. An RO PUF is comprised of many ROs and those to be compared can be both selected in advance or by users with a multiplexer being added before the comparator. Thus the PUF can work in a challenge/response mode. Challenges are ROs chosen by the users to be compared; responses are inputs from comparators.

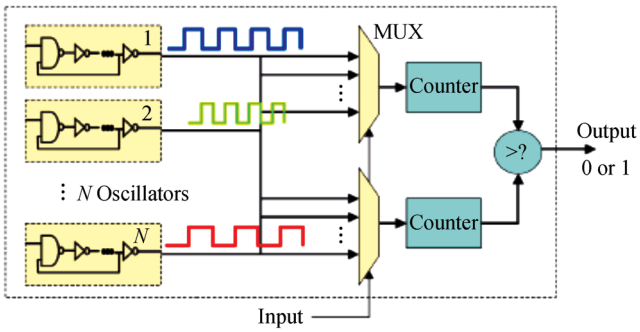


Fig.6. 1-out-of-N ring oscillator PUF architecture[11].

In terms of security, the challenge/response mode is better than a directly formed unique signature. However, the RO PUF can only generate a relatively small number of CRPs (weak PUF), therefore, two methods were proposed to generate more response bits[33]: 1) configure the path within a delay loop so that different challenges result in different oscillation frequencies; 2) each challenge can configure the number of inverters to determine the oscillator configuration to generate a PUF circuit with different numbers of inverters. Maiti *et al.*[34] also proposed an identity-mapping function to increase the number of CRPs for a ring-oscillator PUF (RO-PUF). However, their method does not increase the entropy extracted from a PUF, which means that these CRPs may be dependent with each other.

Silicon PUF is based on manufacture variation, which may be very sensitive to the operating environment such as voltage and temperature, particularly for delay-based PUF[86]. It is very hard for any known PUF to maintain an absolutely stable response. Recently, researchers proposed a lot of methods to enhance the security and improve the reliability of PUF. Vivekraja and Nazhandali[88] studied the effect of operating (supply) voltage of the circuit and the body bias voltage of the transistors in the circuit on the performance of RO PUFs. Compared with a base design, the uniqueness and reproducibility of PUFs can be increased by 18% and 7% respectively by carefully adjusting these two circuit level considerations. Methods such as error correcting[65,75,89], pattern matching[66,74], temperature aware collaboration[90] and configurable method[91-92] have been proposed to correct or avoid bit flips in PUF responses to generate stable PUF out-

put. Škorić *et al.*[93] introduced several methods to reduce the noise at the source and extract as much robust key material as possible by properly choosing an error correction algorithm and therefore finally to improve the robustness of bit-string extraction from noisy PUF measurements. Maiti *et al.*[91-92] proposed a configurable method that challenges select different inverters of a ring oscillator to generate multiple instantiations of ROs inside in a basic configurable RO structure. As shown in Fig.7, a basic four-step configurable RO will generate eight different ROs using the control inputs $c1$, $c2$ and $c3$ of the three 2:1 multiplexers. To improve the reliability of the RO PUF, they can select the pair which has the maximum difference in frequency to generate stable PUF response.

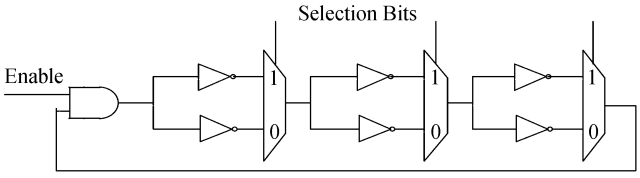


Fig.7. Four-step configurable RO.

Recently, Yin and Qu[90] proposed a temperature-aware cooperative approach with high hardware efficiency to convert the unreliable bits into reliable ones under temperature variations. The conversion is assisted by the neighbor reliable bits. Compared with the common approach using redundancy to provide reliability[33], their method can significantly improve the efficiency of RO PUF implementation.

In traditional pairwised comparison method, one RO PUF bit is generated through the frequency comparison between two ring oscillators. Thus n ROs will generate $n/2$ bits. For the neighbor chain approach[75], n ROs will generate $n-1$ bits. But the number of bits generated by the both methods are limited to the linear upper bound $O(n)$. Yin *et al.*[94-95] proposed to improve this upper bound to $O(n\log_2^n)$ by grouping the ROs under given conditions. Suppose there are four ROs, $\{A, B, C, D\}$. For the pairwised comparison method, four ROs can only generate two bits. However, given four elements, there are totally $4! = 24 = 10111_2$ different permutations. If we put A, B, C and D in one group, all the possible permutations are listed in Fig.8.

00000_2	$\{ABCD\}$	01000_2	$\{BCAD\}$	10000_2	$\{CDAB\}$
00001_2	$\{ABDC\}$	01001_2	$\{BCDA\}$	10001_2	$\{CDBA\}$
00010_2	$\{ACBD\}$	01010_2	$\{BDAC\}$	10010_2	$\{DABC\}$
00011_2	$\{ACDB\}$	01011_2	$\{BDCA\}$	10011_2	$\{DACB\}$
00100_2	$\{ADBC\}$	01100_2	$\{CABD\}$	10100_2	$\{DBAC\}$
00101_2	$\{ADCB\}$	01101_2	$\{CADB\}$	10101_2	$\{DBCA\}$
00110_2	$\{BACD\}$	01110_2	$\{CBAD\}$	10110_2	$\{DCAB\}$
00111_2	$\{BADC\}$	01111_2	$\{CBDA\}$	10111_2	$\{DCBA\}$

Fig.8. Compact syndrome coding (CSC) table.

If we continuously encode each permutation with 5-bit, from 00000 to 10111 by compact syndrome coding (CSC), shown in Fig.8, we will have five independent bits. Given n ROs, there are $n!$ permutations. These permutations can be encoded into $\log_2 n! = n\log_2 n$ bits. In order to guarantee the reliability, the threshold R_{th} needs to be set. The frequency difference between every element in one group should not be smaller than R_{th}.

The partition of the group is determined by the frequencies differences of the ROs. These frequencies differences are affected by two types of variations: process variations and systematic variations. The process variation is the naturally occurring variation in attributes of transistors (length, width, oxide thickness) when integrated circuits are fabricated. The systematic variations are the spatial trends which are easily tested, as shown in Fig.9. The mechanism of PUF is to utilize the random process variations. However these two variations are mixed up. Therefore, Yin and Qu[35] designed a variations distiller using polynomial regression to decompose the variations. The main idea is to use the polynomial curve fitting the trend of the systematic variation. Fig.10 clearly illustrates the distribution and the trend of process variations and systematic variations. The roughness of the surface represents the pro-

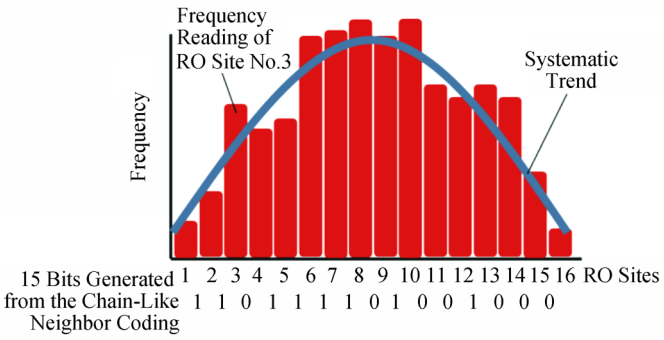


Fig.9. Illustration of the impact from systematic variation even after decoupling.

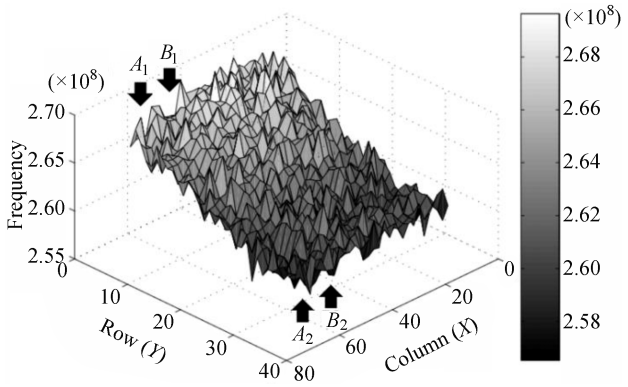


Fig.10. Across-die frequency topology of an RO array.

cess variations while the slop represents the systematic variations. Fig.11 shows the variations after filtering out the systematic trend.

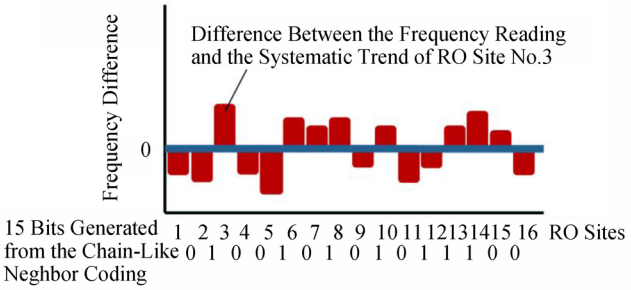


Fig.11. Distilled random fabrication variation after the systematic trend is removed.

The reliability depends on not only how reliable the bits are, but also how efficient the Error Correction Code (ECC) is. Because in some severe environments, it is very difficult to keep all bits stable. Hence, the ECC is usually used. However, the encoding approach, shown in Fig.8, does not work well with ECC. For example, there are four ROs, $\{A, B, C, D\}$, listed from the fastest to the slowest, encoding as 00000_2. With the environment changes, if C becomes the fastest RO, the order of these four ROs becomes $\{C, A, B, D\}$, which is encoded as 01100_2. Compared with initial bits, a small change of the order may effect severe flipping of the final bits, which would be a heavy workload and even out of error correction's ability. Hence, Yin and Qu[89] proposed a KSC (Kendall Syndrome Coding) encoding approach. Compared with CSC, KSC is much more efficient for error correction. KSC decomposes a short and compacted bitstream into a long and spatial bitstream. Rather than severe changes of CSC, minority of KSC bits are affected by the listed example, which is much more convenient for ECC. Even though the KSC bitstream is much longer than the CSC bitstream, their entropy is the same because the bits generated by KSC are dependent with each other, which reduces the security of PUF. Hence, we use the KSC for error correction, and then KSC bits are compacted into the CSC.

Most recently, Gao *et al.* proposed a highly flexible configurable RO PUF[96] to improve the reliability of traditional RO PUF. Compared with the traditional RO PUF[33] which builds on RO level, the new architecture is built at inverter level, shown in Fig.12. This resolution improvement enhances the flexibility of RO PUF. The flexibility is implemented by the multiplexers. The selection bits are controlled to select/bypass the inverter. The main unit is demonstrated in Fig.13. The left cell is considered as a delay unit. Ideally the delay of the delay unit will be either 0 or the delay of the inverter. However, in practice, the path delay going through the multiplexer cannot be ignored. We modify the two delays of path "1" and "0" as two buffers and pull them out of the multiplexer. After filtering these two delays, the multiplexer becomes "ideal". The delays of two paths are named as d_1 and d_0, and the delay difference between two paths is $d_{\text{diff}} = d_i + d_1 - d_0$. Now we can consider the delay of this unit as either 0 (when the selection bit is 0) or d_{diff} (when the selection bit is "1"). Fig.14 shows the measurement schematic, which is inspired by the approach proposed by Majzoobi *et al.*[48] By turning on (selection bit is 1) each delay unit in sequence, we can measure d_{diff} of each unit. The enhancement of flexibility increases the reliability and security for PUF design. Consider the following example. Assume that the delays of these inverters are RO_1: {5, 6, 8, 7, 4} and RO_2: {7, 5, 7, 5, 4}. The total delay D_{RO_1} and D_{RO_2} of these two ROs are 30 and 28,

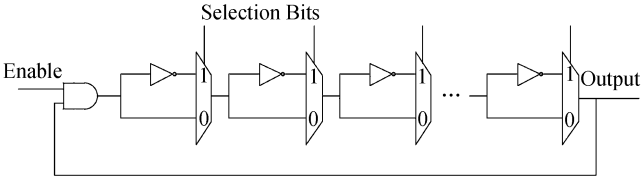


Fig.12. Architecture of the configurable RO.

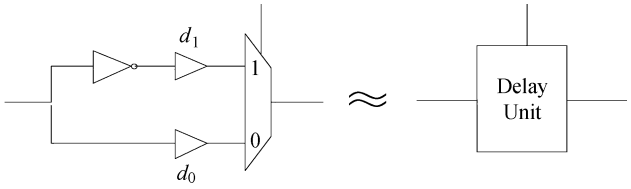


Fig.13. Basic cell and its equivalent model.

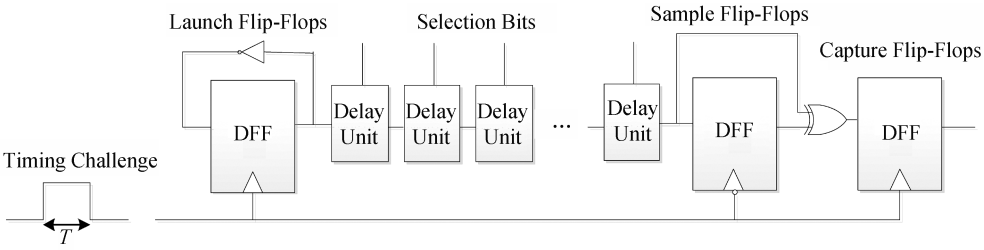


Fig.14. Measurement schematic of flexible RO PUF.

respectively. The delay difference between them is $30 - 28 = 2$. However, the high flexible architecture gives us the opportunity to pick the inverters rather than the whole RO. We can pick {6, 7, 8} for RO_1 and {5, 7, 5} for RO_2. Now the delay of these two new configured RO are 21 and 17 respectively. And the delay difference is $21-17 = 4$ much bigger than 2. As the delay difference becomes bigger, the reliability increases.

Besides the pairwised comparison method, the flexible configurable architecture also can dramatically enhance the efficiency of group-based approach. For example, considering the following three ROs: $RO_1 = \{4, 5, 3, 7, 6\} = 25$; $RO_2 = \{5, 2, 8, 6, 6\} = 27$; $RO_3 = \{6, 3, 4, 7, 4\} = 24$. Assume the threshold R_{th} is set to 3, the partition would be $\{RO_2, RO_3\}$, $\{RO_1\}$. In this case, only one bit is generated. However, if we pick three inverters to form one subring: $sub_1 = \{5, 3, 6\} = 14$ in RO_1; $sub_2 = \{5, 6, 6\} = 17$ in RO_2; $sub_3 = \{3, 4, 4\} = 11$ in RO_3, given $R_{th} = 3$, all these three subrings can be in one group, $\{sub_1, sub_2, sub_3\}$. Thus two bits are generated. Because we can control whether to select (or bypass) any given inverter, we can maximize the number of elements in one group, and generate much more bits. The results in [96] show that the highly flexible architecture achieves the improvement in both security and reliability.

7 Recent Challenges and New Opportunities

We believe it is probably early and inaccurate to claim that we are going to see the fall of PUF. But with the inability of integrating PUF into the practice of trusted system design and the reports exploiting PUF's security vulnerabilities, if we are unable to find effective solutions to the aforementioned *hardware efficiency*, *security*, and *reliability* challenges, PUF may become another failure before it delivers its promises as a hardware security primitive.

From the practical point of view, even for the most mature RO PUF and SRAM PUF, there are still many unsolved scientific and practical challenges before they become a regular design component and hardware security primitive for the trustworthy computing systems. Meanwhile, like the process other security primitives also have to go through, there are recent reports on the vulnerabilities of PUF and several successful attempts to clone the unclonable PUF information (see Section 4). All the attacks reported are within the past couple of years. Although most of the authors have pointed out potential countermeasures to their proposed attacks, most of these countermeasures do not have high practical value because they incur high hardware overhead or increased design complexity while still cannot provide full security for the PUFs. For example, the countermeasure proposed in [58] has 200% area overhead and the RO array remains vulnerable. One of potential solutions is to build PUF based on the existing on-chip hardware components. We present the following two examples to offer readers reference.

One such example is the on-chip temperature sensors. Since an RO's delay is very sensitive to on-chip temperature and the delay-temperature dependency is well understood. This provides a way to conveniently monitor on-chip temperature by measuring the RO's delay and then calculate the corresponding temperature. We can explore the possibility of utilizing such temperature sensors as the ROs for PUF bits generation. As temperature sensors are normally placed around the spots where the temperature may change rapidly such as the hotspots, the reliability of the PUF bits generated by such temperature sensors will be a big concern.

Another example is the scan chain used for testing. For a circuit to have the scan capability, the D type flip-flops (DFFs) are replaced by scan flip-flops (SFFs). An SFF consists of a DFF, a multiplexer, and two new signals: scan-data SD and test control TC. SFFs can be chained by connecting the Q output of one SFF to the SD input of the next SFF. We call this the Q-SD connection style. TC is used to switch the CUT (core under test) between the normal mode and the testing mode. In the normal mode, the SFFs will serve as the normal DFFs. In the testing mode, test data from the new primary input SI will be supplied to the scan chain and the response data will be collected from the new primary output SO. The DFF contained in the SFF normally has another output Q' (also called QN), which is the complement of its output Q. This enables another connection style which connects Q', instead of Q, to the SD of the next SFF. We refer to this as Q'-SD connection between two adjacent SFFs. Most technology libraries provide the test cell of SFF with two outputs complementary to each other, enabling scan designs with both connection styles[97]. This structure provides us opportunities of generating PUF information from the scan chain design similar to those methods for VLSI intellectual property protection[98-99]. By reusing the scan chain as PUF circuitry, we can reduce or eliminate the hardware overhead caused by PUF. However, this will pose new challenges in maintaining the performance of the scan chain design (such as the testing power and test vector generation) and the quality (reliability, uniqueness, security, etc.) of the PUF information.

8 Conclusions

There has been more than a decade of intensive study on PUF since it was introduced into the research

community. Among PUFs of different forms, silicon PUFs are of the most interest in terms of fabrication cost and readiness to be integrated to computing and communication devices. In this paper, we have surveyed the current state-of-the-art of silicon PUFs and presented evaluation criteria for them. We also elaborated the known attacks to PUFs and corresponding countermeasures, and then the three typical applications for PUFs are discussed. We highlighted some recent research advances for RO PUFs. Finally, since hardware efficiency, security, and reliability issues of PUFs have brought new challenges, it is urgent to develop effective solutions to address them.

References

[1] Pappu R, Recht B, Taylor J, Gershenfeld N. Physical one-way functions. *Science*, 2002, 297(5589): 2026-2030.

[2] Gassend B, Clarke D, van Dijk M, Devadas S. Silicon physical random functions. In *Proc. the 9th ACM Conference on Computer and Communications Security*, Nov. 2002, pp.148-160.

[3] Gassend B, Clarke D, van Dijk M, Devadas S. Controlled physical random functions. In *Proc. the 18th Annual Computer Security Applications Conference*, Dec. 2002, pp.149-160.

[4] Nithyanand R, Solis J. A theoretical analysis: Physical unclonable functions and the software protection problem. In *Proc. IEEE Symposium on Security and Privacy Workshops*, May 2012, pp.1-11.

[5] Vrijaldenhoven S. Acoustical physical uncloneable functions [Master Thesis]. T.U. Eindhoven, 2004.

[6] Buchanan J D R, Russell P, Cowburn R P *et al.* Forgery: 'Fingerprinting' documents and packaging. *Nature*, 2005, 436(7050): 475.

[7] Bulens P, Standaert F, Quisquater J. How to strongly link data and its medium: The paper case. *IET Inf. Secur.*, 2010, 4(3): 125-136.

[8] Hammouri G, Dana A, Sunar B. CDs have fingerprints too. In *Proc. the 11th International Workshop on Cryptographic Hardware and Embedded Systems*, Sept. 2009, pp.348-362.

[9] Indeck R, Muller M. Method and apparatus for fingerprinting magnetic media. US Pat. 5365586, 1994.

[10] DeJean G, Kirovski D. RF-DNA: Radio-frequency certificates of authenticity. In *Proc. the 9th International Workshop on Hardware and Embedded Systems*, Sept. 2007, pp.346-363.

[11] Jiang D, Chong C N. Anti-counterfeiting using phosphor PUF. In *Proc. the 2nd International Conference on Anti-counterfeiting, Security and Identification*, Aug. 2008, pp.59-62.

[12] Chong C N, Jiang D, Zhang J, Guo L. Anti-counterfeiting with a random pattern. In *Proc. the 2nd International Conference on Emerging Security Information, Systems and Technologies*, Aug. 2008, pp.146-153.

[13] Tuyls P, Skoric B, Stallinga S, Akkermans T, Ophey W. An information theoretic model for physical uncloneable functions. In *Proc. International Symposium on Information Theory*, Jun. 2004, p.139.

[14] Tuyls P, Škorić B, Stallinga S, Akkermans A H M, Ophey W. Information-theoretic security analysis of physical uncloneable functions. In *Proc. the 9th International Conference on Financial Cryptography and Data Security*, Feb. 28-Mar. 3, 2005, pp.141-155.

[15] Ignatenko T, Schrijen G, Skoric B, Tuyls P, Willems F. Estimating the secrecy-rate of physical unclonable functions with the context-tree weighting method. In *Proc. IEEE International Symposium on Information Theory*, July 2006, pp.499-503.

[16] Lofstrom K, Daasch W R, Taylor D. IC identification circuit using device mismatch. In *Proc. IEEE International Solid-State Circuits Conference*, Feb. 2000, pp.372-373.

[17] Tuyls P, Schrijen G, Škorić B *et al.* Read-proof hardware from protective coatings. In *Proc. the 8th International Workshop on Cryptographic Hardware and Embedded Systems*, Oct. 2006, pp.369-383.

[18] Puntin D, Stanzione S, Iannaccone G. CMOS unclonable system for secure authentication based on device variability. In *Proc. the 34th European Solid-State Circuits Conference*, Sep. 2008, pp.130-133.

[19] Guajardo J, Škorić B, Tuyls P, Kumar S S, Bel T, Blom A H M, Schrijen G. Anti-counterfeiting, key distribution, and key storage in an ambient world via physical unclonable functions. *Information Systems Frontiers*, 2009, 11(1): 19-41.

[20] Helinski R, Acharyya D, Plusquellic J. A physical unclonable function defined using power distribution system equivalent resistance variations. In *Proc. the 46th ACM/IEEE Design Automation Conference*, July 2009, pp.676-681.

[21] Škorić B, Maubach S, Kevenaar T, Tuyls P. Information-theoretic analysis of capacitive physical unclonable functions. *J. Appl. Phys.*, 2006, 100(2): Article No. 024902.

[22] Holcomb D E, Burleson W P, Fu K. Power-up SRAM state as an identifying fingerprint and source of true random numbers. *IEEE Transactions on Computers*, 2009, 58(9): 1198-1210.

[23] Guajardo J, Kumar S S, Schrijen G, Tuyls P. FPGA intrinsic PUFs and their use for IP protection. In *Proc. the 9th International Workshop on Cryptographic Hardware and Embedded Systems*, Sept. 2007, pp.63-80.

[24] Kumar S S, Guajardo J, Maes R, Schrijen G, Tuyls P. Extended abstract: The butterfly PUF protecting IP on every FPGA. In *Proc. IEEE International Workshop on Hardware-Oriented Security and Trust*, Jun. 2008, pp.67-70.

[25] Krishna A, Narasimhan S, Wang X *et al.* MECCA: A robust low-overhead PUF using embedded memory array. In *Proc. the 13th International Workshop on Cryptographic Hardware and Embedded Systems*, Sept. 28-Oct. 1, 2011, pp.407-420.

[26] Zheng Y, Krishna A, Bhunia S. ScanPUF: Robust ultralow-overhead PUF using scan chain. In *Proc. the 18th Asia and South Pacific Design Automation Conference*, Jan. 2013, pp.626-631.

[27] Maes R, Tuyls P, Verbauwhede I. Intrinsic PUFs from flip-flops on reconfigurable devices. In *Proc. the 3rd Benelux Workshop on Information and System* (*WISSec*), Nov. 2008, pp.1-17.

[28] Katzenbeisser S, Koçabas Ü, Rožić V *et al.* PUFs: Myth, fact or busted? A security evaluation of physically unclonable functions (PUFs) cast in silicon. In *Proc. the 14th International Workshop on Cryptographic Hardware and Embedded Systems*, Sept. 2012, pp.283-301.

[29] Helfmeier C, Boit C, Nedospasov D, Seifert J. Cloning physically unclonable functions. In *Proc. IEEE International Symposium on Hardware-Oriented Security and Trust* (*HOST*), Jun. 2013, pp.1-6.

[30] Lee J W, Lim D, Gassend B, Suh G E, van Dijk M, Devadas S. A technique to build a secret key in integrated circuits for identification and authentication applications. In *Proc. Symposium on VLSI Circuits. Digest of Technical Papers*, Jun. 2004, pp.176-179.

[31] Lim D, Lee J, Gassend B, Suh G E, van Dijk M, Devadas S. Extracting secret keys from integrated circuits. *IEEE Trans. Very Large Scale Integr. Syst.*, 2005, 13(10): 1200-1205.

[32] Lin L, Srivathsa S, Krishnappa D K, Shabadi P, Burleson W. Design and validation of arbiter-based PUFs for sub-45-nm low-power security applications. *IEEE Trans. Inf. Forensics Secur.*, 2012, 7(4): 1394-1403.

[33] Suh G E, Devadas S. Physical unclonable functions for device authentication and secret key generation. In *Proc. the 44th ACM/IEEE Design Automation Conference*, Jun. 2007, pp.9-14.

[34] Maiti A, Kim I, Schaumont P. A robust physical unclonable function with enhanced challenge-response set. *IEEE Trans. Inf. Forensics Secur.*, 2012, 7(1): 333-345.

[35] Yin C, Qu G. Improving PUF security with regression-based distiller. In *Proc. the 50th Annual Design Automation Conference*, May 29-Jun. 7, 2013, pp.1-6.

[36] Shimizu K, Suzuki D, Kasuya T. Glitch PUF: Extracting information from usually unwanted glitches. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, 2012, E95-A(1): 223-233.

[37] Anderson J H. A PUF design for secure FPGA-based embedded systems. In *Proc. the 15th Asia and South Pacific Design Automation Conference*, Jan. 2010, pp.1-6.

[38] Aarestad J, Ortiz P, Acharyya D, Plusquellic J. HELP: A hardware-embedded delay PUF. *IEEE Des. Test*, 2013, 30(2): 17-25.

[39] Nithyanand R, Sion R, Solis J. POSTER: Making the case for intrinsic personal physical unclonable functions (IP-PUFs). In *Proc. the 18th ACM Conference on Computer and Communications Security*, Oct. 2011, pp.825-828.

[40] Rührmair U, Sehnke F, Sölter J, Dror G, Devadas S, Schmidhuber J. Modeling attacks on physical unclonable functions. In *Proc. the 17th ACM Conference on Computer and Communications Security*, Oct. 2010, pp.237-249.

[41] Delvaux J, Verbauwhede I. Side channel modeling attacks on 65nm arbiter PUFs exploiting CMOS device noise. In *Proc. IEEE Int. Symposium on Hardware-Oriented Security and Trust*, Jun. 2013, pp.137-142.

[42] Rührmair U, Sölter J, Sehnke F, Xu X, Mahmoud A, Stoyanova V, Dror G, Schmidhuber J, Burleson W, Devadas S. PUF modeling attacks on simulated and silicon data. *IEEE Trans. Inf. Forensics Secur.*, 2013, 8(11): 1876-1891.

[43] Saha I, Jeldi R R, Chakraborty R S. Model building attacks on physically unclonable functions using genetic programming. In *Proc. IEEE International Symposium on Hardware-Oriented Security and Trust*, Jun. 2013, pp.41-44.

[44] Gassend B, Lim D, Clarke D, van Dijk M, Devadas S. Identification and authentication of integrated circuits. *Concurr. Comput. Pract. Exp.*, 2004, 16(11): 1077-1098.

[45] Majzoobi M, Koushanfar F, Potkonjak M. Testing techniques for hardware security. In *Proc. IEEE International Test Conference*, Oct. 2008.

[46] Majzoobi M, Koushanfar F, Potkonjak M. Lightweight secure PUFs. In *Proc. IEEE/ACM International Conference on Computer-Aided Design*, Nov. 2008, pp.670-673.

[47] Morozov S, Maiti A, Schaumont P. An analysis of delay based PUF implementations on FPGA. In *Proc. the 6th International Symposium on Applied Reconfigurable Computing*, Mar. 2010, pp.382-387.

[48] Majzoobi M, Koushanfar F. Devadas S. FPGA PUF using programmable delay lines. In *Proc. IEEE International Workshop on Information Forensics and Security*, Dec. 2010.

[49] Ozturk E, Hammouri G, Sunar B. Physical unclonable function with tristate buffers. In *Proc. IEEE International Symposium on Circuits and Systems*, May 2008, pp.3194-3197.

[50] Zhang J, Wu Q, Lyu Y, Zhou Q, Cai Y, Lin Y, Qu G. Design and implementation of a delay-based PUF for FPGA IP protection. In *Proc. IEEE International Conference on Computer-Aided Design and Computer Graphics*, Nov. 2013, pp.107-114.

[51] Dodis Y, Ostrovsky R, Reyzin L, Smith A. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 2008, 38(1): 97-139.

[52] Karakoyunlu D, Sunar B. Differential template attacks on PUF enabled cryptographic devices. In *Proc. IEEE International Workshop on Information Forensics and Security*, Dec. 2010.

[53] Dai J, Wang L. A study of side-channel effects in reliability-enhancing techniques. In *Proc. the 24th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*, Oct. 2009, pp.236-244.

[54] Merli D, Schuster D, Stumpf F, Sigl G. Side-channel analysis of PUFs and fuzzy extractors. In *Proc. the 4th International Conference on Trust and Trustworthy Computing*, Jun. 2011, pp.33-47.

[55] Schuster D. Side-channel analysis of physical unclonable functions (PUFs) [Master Thesis]. Technische Universität Wien, 2010.

[56] Merli D, Schuster D, Stumpf F, Sigl G. Semi-invasive EM attack on FPGA RO PUFs and countermeasures. In *Proc. Workshop on Embedded Systems Security*, Oct. 2011, Article No. 2.

[57] Mahmoud A, Rührmair U, Majzoobi M, Koushanfar F. Combined modeling and side channel attacks on strong PUFs. *IACR Cryptology ePrint Archive*, Article No. 632, 2013.

[58] Merli D, Heyszl J, Heinz B, Schuster D, Stumpf F, Sigl G. Localized electromagnetic analysis of RO PUFs. In *Proc. IEEE International Symposium on Hardware-Oriented Security and Trust*, Jun. 2013, pp.19-24.

[59] Merli D, Stumpf F, Sigl G. Protecting PUF error correction by codeword masking. *IACR Cryptology ePrint Archive*, Article No. 334, 2013.

[60] Gassend B, Van Dijk M, Clarke D, Torlak E, Devadas S, Tuyls P. Controlled physical random functions and applications. *ACM Trans. Inf. Syst. Secur.*, 2008, 10(4): Article No. 3.

[61] Guajardo J, Kumar S S, Schrijen G, Tuyls P. Brand and IP protection with physical unclonable functions. In *Proc. IEEE International Symposium on Circuits and Systems*, May 2008, pp.3186-3189.

[62] Zhang J, Lin Y, Lyu Y, Qu G, Cheung R C C, Che W, Zhou Q, Bian J. FPGA IP protection by binding finite state machine to physical unclonable function. In *Proc. the 23rd International Conference on Field Programmable Logic and Applications*, Sept. 2013.

[63] Zhang J, Lin Y, Lyu Y, Cheung R C C, Che W, Zhou Q, Bian J. Binding hardware IPs to specific FPGA device via intertwining the PUF response with the FSM of sequential circuits. In *Proc. the 21st IEEE Annual International Symposium on Field-Programmable Custom Computing Machines*, Apr. 2013, p.227.

[64] Guajardo J, Kumar S S, Schrijen G, Tuyls P. Physical unclonable functions and public-key crypto for FPGA IP protection. In *Proc. International Conference on Field Programmable Logic and Applications*, Aug. 2007, pp.189-195.

[65] Maes R, Van Herrewege A, Verbauwhede I. PUFKY: A fully functional PUF-based cryptographic key generator. In *Proc. the 14th International Workshop Cryptographic Hardware and Embedded Systems*, Sept. 2012, pp.302-319.

[66] Majzoobi M, Rostami M, Koushanfar F, Wallach D S, Devadas S. Slender PUF protocol: A lightweight, robust, and secure authentication by substring matching. In *Proc. IEEE Symposium on Security and Privacy Workshops*, May 2012, pp.33-44.

[67] Hammouri G, Öztürk E, Birand B, Sunar B. Unclonable lightweight authentication scheme. In *Proc. the 10th International Conference on Information and Communications Security*, Oct. 2008, pp.33-48.

[68] Majzoobi M, Koushanfar F. Time-bounded authentication of FPGAs. *IEEE Trans. Inf. Forensics Secur.*, 2011, 6(3): 1123-1135.

[69] Öztürk E, Hammouri G, Sunar B. Towards robust low cost authentication for pervasive devices. In *Proc. the 6th IEEE International Conference on Pervasive Computing and Communications*, Mar. 2008, pp.170-178.

[70] Alkabani Y, Koushanfar F. Active control and digital rights management of integrated circuit IP cores. In *Proc. International Conference on Compilers, Architectures and Synthesis for Embedded Systems*, Oct. 2008, pp.227-234.

[71] Koushanfar F. Provably secure active IC metering techniques for piracy avoidance and digital rights management. *IEEE Trans. Inf. Forensics Secur.*, 2012, 7(1): 51-63.

[72] Suh G E, O'Donnell C W, Devadas S. Aegis: A single-chip secure processor. *IEEE Des. Test Comput.*, 2007, 24(6): 570-580.

[73] Asim M, Guajardo J, Kumar S S, Tuyls P. Physical unclonable functions and their applications to vehicle system security. In *Proc. the 69th IEEE Vehicular Technology Conference*, Apr. 2009.

[74] Paral Z S, Devadas S. Reliable and efficient PUF-based key generation using pattern matching. In *Proc. IEEE International Symposium on Hardware-Oriented Security and Trust*, Jun. 2011, pp.128-133.

[75] Yu M, Devadas S. Secure and robust error correction for physical unclonable functions. *IEEE Des. Test Comput.*, 2010, 27(1): 48-65.

[76] Anderson R, Kuhn M. Low cost attacks on tamper resistant devices. In *Proc. the 5th International Workshop on Security Protocols*, Apr. 1998, pp.125-136.

[77] Kocher P, Jaffe J, Jun B. Differential power analysis. In *Proc. the 19th Advances in Cryptology,* Aug. 1999, pp.388-397.

[78] Zhang J, Wu Q, Chen J. Research on design method of dynamic partial reconfigurable system. *J. Softw. Eng.*, 2012, 6(2): 21-30.

[79] Gora M A, Maiti A, Schaumont P. A flexible design flow for software IP binding in FPGA. *IEEE Trans. Industrial Informatics*, 2010, 6(4): 719-728.

[80] Zhang J, Lin Y, Wu Q, Che W. Watermarking FPGA bitfile for intellectual property protection. *Radioengineering*, 2012, 21(2): 764-771.

[81] Zhang J, Lin Y, Che W, Wu Q, Lu Y, Zhao K. Efficient verification of IP watermarks in FPGA designs through lookup table content extracting. *IEICE Electron. Express*, 2012, 9(22): 1735-1741.

[82] Zhang J, Lin Y, Lyu Y, Wang X. A chaotic-based publicly verifiable FPGA IP watermark detection scheme. *Sci. CHINA Inf. Sci.*, 2013, 43(9): 1096-1110.

[83] Maes R, Schellekens D, Verbauwhede I. A pay-per-use licensing scheme for hardware IP cores in recent SRAM-based FPGAs. *IEEE Trans. Inf. Forensics Secur.*, 2012, 7(1): 98-108.

[84] Guneysu T, Moller B, Paar C. Dynamic intellectual property protection for reconfigurable devices. In *Proc. International Conference on Field-Programmable Technology*, Dec. 2007, pp.169-176.

[85] Kepa K, Morgan F, Kosciuszkiewicz K. IP protection in partially reconfigurable FPGAs. In *Proc. International Conference on Field Programmable Logic and Applications*, Aug. 2009, pp.403-409.

[86] Zhang J, Lin Y, Lyu Y, Qu G. A PUF-FSM binding scheme for FPGA IP protection and pay-per-device licensing. *IEEE Trans. Inf. Forensics Secur.*, 2014. (to be appeared)

[87] Maes R, Tuyls P, Verbauwhede I. A soft decision helper data algorithm for SRAM PUFs. In *Proc. IEEE International Symposium on Information Theory*, Jun. 28-Jul. 3, 2009, pp.2101-2105.

[88] Vivekraja V, Nazhandali L. Circuit-level techniques for reliable physically uncloneable functions. In *Proc. IEEE International Workshop on Hardware-Oriented Security and Trust*, Jul. 2009, pp.30-35.

[89] Yin C E, Qu G. Kendall syndrome coding (KSC) for group-based ring-oscillator physical unclonable functions. Technical Report, ISR Technical Report 2011-13, 2011, http://drum.lib.umd.edu/handle/1903/12158, May 2014.

[90] Yin C E, Qu G. Temperature-aware cooperative ring oscillator PUF. In *Proc. IEEE International Workshop on Hardware-Oriented Security and Trust*, Jul. 2009, pp.36-42.

[91] Maiti A, Schaumont P. Improved ring oscillator PUF: An FPGA-friendly secure primitive. *Journal of Cryptology*, 2011, 24(2): 375-397.

[92] Maiti A, Schaumont P. Improving the quality of a physical unclonable function using configurable ring oscillators. In *Proc. the 19th International Conference on Field-Programmable Logic and Applications*, Aug. 31-Sept. 2, 2009, pp.703-707.

[93] Škorić B, Tuyls P, Ophey W. Robust key extraction from physical uncloneable functions. In *Proc. the 3rd International Conference on Applied Cryptography and Network Security*, Jun. 2005, pp.407-422.

[94] Yin C E, Qu G. LISA: Maximizing RO PUF's secret extraction. In *Proc. IEEE International Symposium on Hardware-Oriented Security and Trust*, Jun. 2010, pp.100-105.

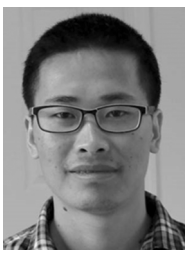
[95] Yin C E, Qu G, Zhou Q. Design and implementation of a group-based RO PUF. In *Proc. Design, Automation & Test in Europe Conference & Exhibition*, Mar. 2013, pp.416-421.

[96] Gao M, Lai K, Qu G. A highly flexible ring oscillator PUF. In *Proc. the 51st ACM/IEEE Design, Automation Conference*, Jun. 2014.

[97] Gupta S, Vaish T, Chattopadhyay S. Flip-flop chaining architecture for power-efficient scan during test application. In *Proc. the 14th Asian Test Symposium*, Dec. 2005, pp.410-413.

[98] Cui A, Chang C H. An improved publicly detectable watermarking scheme based on scan chain ordering. In *Proc. IEEE Int. Symp. Circuits Syst.*, May 2009, pp.29-32.

[99] Chang C, Cui A. Synthesis-for-testability watermarking for field authentication of VLSI intellectual property. *IEEE Trans. Circuits Syst. I Regul. Pap.*, 2010, 57(7): 1618-1630.



Ji-Liang Zhang received the B.E. degree from Shandong University of Science and Technology, Qingdao, in 2009. In 2013∼2014, he works as a research scholar at the Maryland Embedded Systems and Hardware Security Lab, Department of Electrical and Computer Engineering, University of Maryland, College Park. He is currently working toward his Ph.D. degree in the College of Information Science and Engineering, Hunan University. His research interests include hardware security and trust such as security for FPGAs, PUF & PUF-related applications, IC obfuscation, IP protection, and trusted computing.

Gang Qu received the B.S. and M.S. degrees in mathematics from the University of Science and Technology of China, Hefei, in 1992 and 1994, respectively, and the Ph.D. degree in computer science from the University of California, Los Angeles, in 2000. Upon graduation, he joined the University of Maryland at College Park, where he is currently a professor in the Department of Electrical and Computer Engineering and Institute for Systems Research. He is also a member of the Maryland Cybersecurity Center and the Maryland Energy Research Center. Dr. Qu is the director of Maryland Embedded Systems and Hardware Security Lab and the Wireless Sensors Laboratory. His primary research interests are in the area of embedded systems and VLSI CAD with focus on low power system design and hardware related security and trust. He studies optimization and combinatorial problems and applies his theoretical discovery to applications in VLSI CAD, wireless sensor network, bioinformatics, and cybersecurity. Dr. Qu has received many awards for his academic achievements, teaching, and service to the Research community. He is a senior member of IEEE and serving as associate editor for the IEEE Transactions on Computers, IEEE Embedded Systems Letters and Integration, the VLSI Journal.

Yong-Qiang Lv received the B.S. degree in computer science from Xi Dian University in 2001, the M.S. and the Ph.D. degrees in computer science from Tsinghua University in 2003 and 2006 respectively. Currently he is an assistant professor in the Research Institute of Information Technology at Tsinghua University. His research interest focuses on the hardware-software fusion architecture in emerging computing systems, such as Internet of Things, mobile computing, and reconfigurable computing.

Qiang Zhou received his B.S. degree in computer science from University of Science and Technology of China, Hefei, in 1983, M.S. degree in computer science from Tsinghua University, Beijing, in 1986, and Ph.D. degree in control theory and control engineering from Chinese University of Mining and Technology, Beijing, in 2002. He has been a professor in EDA Lab of the Department of Computer Science and Technology, Tsinghua University. His research interests include VLSI layout algorithms and systems.