Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Data Remanence Based Approach towards Stable Key Generation from Physically Unclonable Function Response of Embedded SRAMs using Binary Search

Pyi Phyo Aung[1], Nordinah Ismail[1,*], Chia Yee Ooi[1], Koichiro Mashiko[1], Hau Sim Choo[1], Takanori Matsuzaki[2]

[1] Malaysia–Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia, Jalan Sultan Yahya Petra, 54100 Kuala Lumpur, Malaysia
[2] Dept. of Electric and Electronic Engineering, Faculty of Humanity-Oriented Science and Engineering, Kindai University, 11-6 Kayanomori, Iizuka, Fukuoka 820-8555 Japan

ARTICLE INFO

Article history:
Received 1 July 2023
Received in revised form 22 November 2023
Accepted 1 December 2023
Available online 31 December 2023

Keywords:

Physically unclonable function; binary search; data remanence; authentication

ABSTRACT

Today's device authentications in IoT devices use public and private key cryptography. Nevertheless, they are still vulnerable to threats because keys or device IDs digitally stored in IoT devices can be stolen or cloned. In contrast, SRAM PUFs utilize physical variations in memory cells of embedded SRAM in microcontrollers or standalone SRAM chips. These inherent physical characteristics are unpredictable and practically impossible to duplicate. They are negligible to affect regular SRAM operation but large enough to be used for authentication purposes in SRAM PUFs operation. However, SRAM PUFs have poor stability and a relatively high bit error rate (BER). Temporal Majority Voting (TMV) and other error correction codes (ECCs) have improved SRAM PUFs performance, but they require a lot of processing time and hardware resources. The data remanence nature of SRAM cells can be utilized to select SRAM PUFs bits with much lower BER and more stable bits, but a suitable algorithm is required to find the best possible power-off time for each type of chip. This paper proposes using the data remanence method and binary search algorithm to obtain the strong SRAM PUFs characteristics of the selected SRAMs at the optimal power-off time. These SRAMs include embedded SRAMs of AtMega328P, STM32F108C, ESP8266 microcontrollers, and an off-the-shelf SRAM chip 23LC1024-I/P, which are being used in various IoT applications. The strong SRAM PUF has more stable characteristics that reduce BER to 0% and increase stability to 99.999%. This proposed method can be utilized on any IoT platform which deals with essential data and requires less resource-hungry security and authentication protocols.

* *Corresponding author.*
E-mail address: nordinah.kl@utm.my

https://doi.org/10.37934/araset.35.2.114131

1. Introduction

In 2020, the number of IoT connections surpassed non-IoT connections for the first time. It is estimated that by 2025, the number of electronic gadgets connected to the internet will be more than 30 billion [1]. It is predicted that in 2030, the adoption rate of IoT devices can reach up to 176% [2] and the market for IoT devices will fast become one of the major consumer electronic markets. Since most IoT devices detect and store their users' confidential data, the security of these devices is highly essential [3]. Moreover, the authenticity of each device is prone to several threats because most of the IoT devices are based on small and inexpensive processing chips with little consideration of security challenges [4]. Not only that but there have also been several security breach cases due to the improper and unreliable security protocols of IoT devices.

Existing encryption and authentication systems use private and public-key cryptography [5]. The host devices compute key generation in those systems, and the security protocols must be saved in non-volatile memory. However, most IoT devices usually do not have enough computing power to generate adequate security keys on their own. So, it is challenging to design and construct a reliable and secure IoT system using the current security and authentication protocols. To solve those problems, Physically Unclonable Functions (PUFs) are recommended. PUFs are the digital fingerprints and unique identifications of semiconductor devices. They can be applied in security, authentication [6], and attestation of chips and devices. They provide unique identities to such devices while their designs are identical. PUFs are based on physical imperfections and variations which occur naturally during the manufacturing process of semiconductor devices. Since they are based on physical variations of individual devices, they are practically impossible to predict or duplicate. Therefore, they can provide significant security measures without consuming too many resources and energy from the host device. PUFs are being used in some microcontrollers and FPGAs as lightweight security solutions and are included in many authentication processes of such systems [7]. Due to their usefulness, PUFs can also be used for secure communication protocols [8].

There are many different types of PUFs being researched. They can be classified into several categories using a categorization scheme based on their application, source of randomness (being implicit or explicit), family, and concept [9]. One notable type of PUF is SRAM PUF [10]. They are memory-based PUFs which can exploit the SRAMs embedded in simple devices such as microcontrollers or standalone off-the-shelf SRAM chips [11, 12]. This paper proposes to develop an improved-performance security and authentication system for IoT applications by using SRAM-based PUFs.

2. Previous Works

Lipps *et al*., [13] used AtMega2560 microcontrollers to find the entropy of SRAM PUF and correlated them with external influencing factors environmental temperature or the supply voltage. They claimed that the AtMega2560 MCU is well suited for security and authentication purposes. In 2019, Babaei and Schiele [14] investigated several IoT platforms that utilize PUF and their computational resources and energy concerns. They concluded that the PUF remains an active research area with many challenges to be addressed.

Pehl *et al*., [15] analyzed SRAM PUF on 6-T SRAM cells, transformed them into analog PUF circuits, and performed the robustness benchmarking using 65nm CMOS chips. They verified the randomness, uniqueness, and robustness of their proposed design. Fujiwara *et al*., [16] assessed the error rate, uniqueness, and reliability of 45nm CMOS SRAM PUF under various temperature and voltage conditions. They highlighted the pros and cons of SRAM PUFs.

Using the initial start-up values of SRAM as a digital fingerprint was first proposed in 2007 by Holcomb *et al*., [17]. They utilized the uninitialized values of embedded CMOS SRAM from virtual tags, microcontrollers, and WISP UHF RFID tags to generate random numbers for digital fingerprint extraction. They generated 128 bits of random number for cryptography key generation from 256 bytes of SRAM PUF using 160 different circuits. The extracted SRAM PUF was able to comply with various cryptographic tests.

Cortez *et al*., [18] have also analyzed the repeatability and uniqueness of general purpose (GP) and low-power (LP) SRAM PUF designs by using both circuit simulations and measurements. They also analyzed the start-up values (SUVs) of SRAM PUF and their static noise margin (SNM) [19]. Zhang *et al*., [20] proposed to use Fin field effect transistor (FinFET) SRAM PUF instead of conventional CMOS SRAM. They researched the SNM and the reliability of the FinFET SRAM PUFs and concluded that their design has reasonable results. Narasimham *et al*., [21] also researched 28nm and 16nm FinFET SRAM PUFs concluded that their design can withstand aging-related instabilities.

Wang *et al*., [22] explored the prospective applications of SRAM PUF and investigated the instability in SRAM PUFs. They proposed different power-on techniques to improve the reliability of SRAM PUFs for cryptographic operations. Ziyang *et al*., [23] have also used different power-up scenarios and analyzed the results for 256 bits of SRAM PUF. Elshafiey [24] also stated that the start-up values of SRAM PUF are affected by the rising time of power supply. They implemented a 180nm Silicon-Germanium Bipolar/CMOS (BiCMOS) SRAM and confirmed the results.

Van Aubel *et al*., [25] utilized the SRAMs from the registered AMD64 CPUs and Nvidia GPUS to avoid using dedicated external hardware for PUF. They concluded that although the AMD64 CPU registers are found to have non-random non-fingerprinting behavior, Nvidia GPUs provided promising results to be used for PUFs. In 2017, Wilde [26] analyzed the SRAM PUF on 144 Infineon XMC4500 microcontrollers which contain 160 KB of SRAM. They found that the SRAM PUF has average reliability, bit-alias, uniformity and mid-range in uniqueness, confirming the previous results on other microcontrollers.

Takeuchi *et al*., [27] measured SRAM data after power-up for an addressable SRAM cell array and found that the SRAM's address switching noise and memory effect affect the results significantly. They also conducted measurements to facilitate the characterization of SRAM power-up behavior and proposed methods to obtain a reliable and stable power-up state [28]. Midspan *et al*., proposed “Two Choose One” PUF (TCO-PUF) based on a differential architecture and the non-linear relationship between current and voltage. They investigated the robustness of TCO-PUF and compared it with Arbiter-PUF, Ring Oscillator PUF, and SRAM PUF. They concluded that TCO-PUF and Arbiter PUF are less vulnerable to aging [29].

Trujillo *et al*., [30] researched implementing SRAM PUF on silicon-germanium wafers and proved that their circuits are suitable for security purposes by evaluating randomness, hamming distance, uniqueness, and reliability. Zhang *et al*., [31] have done a detailed statistical analysis on SRAM PUFs and discussed the failure rates and the variations in entropy. They also proposed optimization methods for SRAM PUF.

Barbareschi *et al*., [32] tested the 90nm SRAM PUFs on STM32F3 and STM32F4 microcontrollers and analyzed their stability, reliability, uniqueness, uniformity, and effect of temperature and applied voltage. They concluded that SRAM PUFs are suitable for security purposes after applying fuzzy extractors for error elimination. Liao and Guan [33] conducted to validate the assumptions of SRAM PUFs for spatial cell dependency. They recommended that the unwanted dependency effects can be avoided by carefully selecting SRAM cells. Liao *et al*., [34] discussed the Discharge Inversion Effect (DIE) of SRAM chips and pointed out how it can affect the SRAM power-up applications. They

provided some procedures for better data collection. Alheyasat *et al*., [35] performed the mismatch factor analysis on different PUFs and proved the viability of a robust PUF bit selection method.

In 2011, Handschuh [36] reviewed and analyzed the prospective implementation of different PUFs. They focused primarily on SRAM PUFs and their industrial applications. They investigated the characteristics and properties of SRAM PUF under temperatures ranging from -40°C to +150°C. Furthermore, they measured the aging results for up to 25 years and stated that SRAM PUFs have relatively high error rates and traditional error correction methods may not be able to obtain reliable PUFs. They suggested that a combination of several error correction codes will be needed for robust, vital generations. They also presented the industrial applications of SRAM PUFs and their limitations that need to be addressed further.

Concisely, SRAM PUFs have relatively high randomness and uniqueness, moderate stability, and repeatability, reasonably low but non-zero bit-error rate. Moreover, all the researchers agreed that they are suitable for security systems after improving such characteristics. This can be done by applying error correction codes, physically modifying the SRAM cells, or using other stable bit selection methods.

The SRAM cells tend to retain the data stored in them for a short amount of time after the power supply is cut off. Ram *et al*., [37] had researched this phenomenon in detail in 2002. They have done several SRAM experiments and their data remanence properties under various temperatures and power-off times. They concluded that different SRAM chips behave differently about retaining the same amount of data under the same temperature settings. The amount of data retained is greatly affected by the duration of power cut-off and the temperature.

Liu *et al*., [38] proposed using the data remanence nature of SRAMs to find the most stable SRAM PUF bits. First, they wrote all 1s and all 0s to SRAM arrays to test the data remanence. Then, they manipulate the power-off time of SRAMs and record the flipped bits. The cells that flipped with the least amount of power of time are the most robust SRAM PUF cells. They used this technique to find both the strongest 1-cells and strongest 0-cells. They managed to generate a 100% stable SRAM PUF of the 256-bit key from 512 bits initial SRAM values.

Sajim [39] has researched SRAM PUF from off-the-shelf serial SRAM chips using the data remanence method and neighbour analysis. They used Microchip 23LC1024 and the Cypress CY62256NLL serial SRAM chips to test the viability of SRAM PUF on external SRAM chips. They also proposed an application using off-the-shelf SRAM chips as a security system. Using the enrolment-reconstruction mechanism, they generated a PUF authentication key and utilized the PUF generated key for the security application. They also proposed a method using multiple challenge-response pairs (CRPs) from SRAM PUF. They verified their results by implementing it to generate Bitcoin keys [39].

3. Terminology and Performance Metrics

The PUF function are usually described in Challenge-Response Pairs (CRPs) [17]. The response (*R*) is the function (*f*) of the challenge (*C*). Hence, the PUF function can be represented as a Challenge-Response Pairs (CRPs) as shown in Equation (1).

$$R = f(C) \tag{1}$$

Different types of PUFs have different ways of obtaining CRPs. As for the SRAM PUF, challenge *C* can simply be the memory address and the bit location of the addressed word, and the response *R* is the start-up binary value of that particular bit [40]. For instance as shown in Figure 1, if a given

challenge *C* is {0111 0010, 0100} where 0111 0010 is the memory address while 0100 is the bit location (bit 4 from least-significant bit), response *R = f(0111 0010,0100) = 1*.

Alternatively, the uninitialized start-up streams of binary data from the SRAMs (multiple responses in bulk) are first obtained using any range of memory addresses as challenge values *C*. A specialized program is used to obtain the 64 bytes (512 bits) of PUF streams from a range of memory addresses. The obtained PUF response values, *R,* are then stored together with their respective Challenge addresses *Cs* in the computer for further data analysis. The same challenge values (i.e., the same range of memory addresses) are used for the same type of chips, but each chip's responses will be different.

Address	Start Up Value
...	...
0111 0000	0100 1110
0111 0001	0001 0110
0111 0010	1101 0011
0111 0011	0101 0101
0111 0100	1100 1001
0111 0101	0100 1110
0111 0110	0101 0011
0111 0111	1011 1111
...	...

Fig. 1. An example of SRAM start-up content with the corresponding addresses

3.1 Error Rate

PDF response value *R* from each chip will be compared and evaluated through the most represented values via majority voting algorithm (MV). The difference in each bit of the *R* is called the intra-distance D_{intra} [41]. In other words, D_{intra} is the error bit of the response value *R,* and it can be used to get the error rate *E* of the SRAM PUF. As presented in Equation (2), the error rate can be expressed as the fraction of the number of different bits over the total number of n bits in percentage.

$$E = \frac{D_{intra}}{n} \times 100\% \quad (2)$$

The error rate *E* is sometimes abbreviated as Bit Error Rate (BER). This is the most significant characteristic of SRAM PUF to test its repeatability. The BER can determine whether the system is reliable enough to be used for security and or authentication purposes. The ideal error rate is 0%, where all the iterations of PUF produce the exact same binary stream.

3.2 Uniqueness

The uniqueness is the characteristic of SRAM PUF, which shows how different the PUF stream of one chip is from another. It can be represented using the value of inter-distance $D_{inter,}$ which is the difference between two PUF responses of an equal number of bits from different chips but of the same type [41]. That can be evaluated by finding the difference between the response values

obtained from different chips using the same challenge *C*. The formula to calculate D_{inter} is presented in Equation (3).

$$D_{inter}(R_x, R_y) = \Delta(R_x, R_y) \tag{3}$$

Similar to the error rate, the inter-distance of the PUF function can also be represented as the fraction of D_{inter} over the total number of bits *n* [42]. The fractional inter-distance *I* can be calculated using the formula given in Equation (4).

$$I(R_x, R_y) = \frac{\Delta(R_x, R_y)}{n} \tag{4}$$

Uniqueness can be used to ascertain that a PUF stream is unique, and no other chip can produce the same value. In other words, it can be used to quantify how unique an SRAM PUF binary stream for a particular type of system is. It is a critical factor for the identification and authentication of chips. Since the SRAM PUF deals with binary values, 50% with zero standard deviation is ideal.

3.3 Biasness

Another useful characteristic of PUF is the biasness of the PUF stream towards either 0 or 1. It can be calculated as the ratio between the number of 0s and 1s in a stream of bits. It can determine how well-uniformed the PUF stream is. The biasness or the uniformity of PUF is also known as Fractional Hamming Weight *W* and can be calculated using the formula presented in Equation (5).

$$W(R) = \frac{\#\,(i{:}R_i \neq 0)}{n} \tag{5}$$

Biasness is useful for testing the randomness or uniformity of a binary stream. The ideal value of biasness is 50%, where the binary stream is considered to be well-uniformed and biased neither towards '1' nor '0'.

3.4 Randomness

The randomness of a binary stream is useful for benchmarking how random the data is for the random number generation. Randomness can be measured using the Binary Entropy Function. The Binary Entropy Function is a type of the Shannon Entropy, and it ranges from 0 (the least random value) to 1 (the most random value) [43]. It can be denoted as *H(p),* where *p* is the probability of the biasness or Fractional Hamming Weight *W* [44]. The formula for the randomness *H(p)* is presented in Equation (6).

$$H(p) = -p * log_2(p) - (1-p) * log_2(1-p) \tag{6}$$

The graph for *H(p)* for all possible values of *p* is illustrated in Figure 2. The randomness is the highest when the value of *p* is 0.5, which is when the biasness is at its ideal value of 50%. When the PUF response is biased towards 0 or 1, the randomness will be 0.

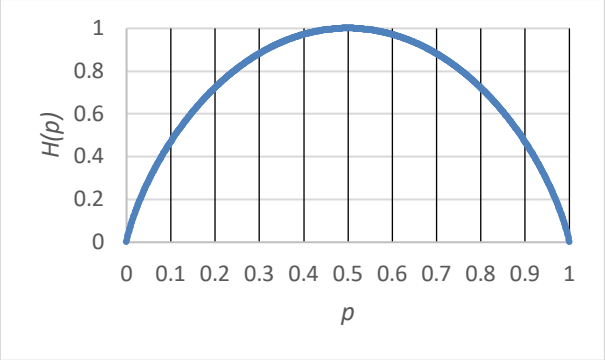

Fig. 2. Graph of *H(p)* for All Values of *p*

3.5 Stability

Stability *S* of the SRAM PUF can be calculated from the ratio between the number of stable bits and the total number of bits. In other words, it is the ratio between the number of bits that never change their values throughout all iterations versus the total number of bits. It is also called the steadiness of the PUF. The stability can be represented as Equation (7).

$$S = \frac{\#\,(i{:}E_i{=}0)}{n} \tag{7}$$

Stability, together with the error rate, can determine a PUF stream's reliability and repeatability. The stability of as close to the ideal stability at 100% as possible is desired in reliable and robust systems.

In this research, the bit error rate *BER*, the uniqueness, the biasness, the randomness, and the stability will be evaluated and analyzed. Those characteristics will then be utilized to find the ideal PUF for each of the microcontrollers and chips. The results will be used to compare and benchmark the error rate reduction and stable bit selection using data remanence and optimal power off time algorithm discussed in the following section

4. A New Data Remanence Method for SRAM PUF by Utilizing Binary Search

In the method to identify the strongest '0' SRAM cells, all the SRAM cells are first written '1's. Next, the SRAM cells will be powered down by stopping the supply voltage for time T_0. The SRAM is then powered on again to enable the connecting microprocessor to record the start-up values of memory cells. Because all the memory arrays are written as 1's, only the strongest '0' cells will flip to their preferred value of '0' while the weak SRAM PUF cells will remain storing the value of '1'. The strongest '1' SRAM cells can be obtained using similar method. It is done by setting all the cells to '0's and then recording the flipped bits after powering off for a short duration. The process must be repeated several times to obtain the most reliable results. This method can reduce testing time to detect strong '0' and strong '1' SRAM cells compared to other methods of extracting SRAM PUF. Figure 3 (a) and (b) illustrate the algorithm to obtain those values.

To use the data remanence method for stable bit selection in SRAM PUFs, the duration of the power-on time and the power-off time must be controlled precisely and independently. To achieve this, an additional microcontroller is added between the microcontroller under test and the computer for precise timing and power controls. An electronic switch is used instead of an electromechanical relay to get the most precise timing. Then, 512 bits (64 Bytes) of initial SRAM

values from the microcontrollers and serial SRAM chips were recorded using different power-off time (*T*) to select the strong '0's and strong '1's.

For each *T* value, 20 iterations have been performed, and the data of every chip for each type of microcontroller is stored in the computer for further analysis. Different chips have different values of the power-off time when some of the bits start flipping [38]. The very first few bits which start to flip are the most substantial bits for that value. The memory addresses and the bit number of all most robust 1's and strongest 0's of all the chips are logged for several power-down time. They are then compared with the initial SRAM values for subsequent evaluations.

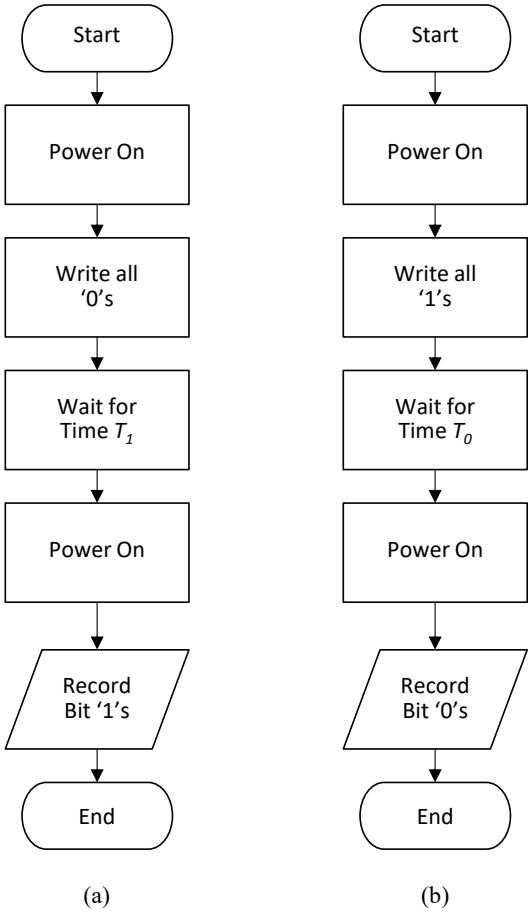

Fig. 3. Algorithm used for obtaining (a) strong '1" bits and (b) strong '0' bits

Binary search, also known as half-interval search, is a type of logarithmic searching algorithm that can find the position of the desired value within sorted arrays much faster than linear search for large arrays with the computational complexity of O (log n). Figure 4 illustrates the way binary search works.

Binary search compares the target value to the center element in an array. If the target value is not found at the center element, that half where there is no target value is eliminated. Then, the search resumes on another half, comparing the new center element with the target value, and repeating this until the target value is found [45].

Since the relationship between the power-off time for SRAMs and the number of flipped bits is almost linear within a specific range, changing the power-off time will be able to control the number of flip bits. For an ideal random binary stream of PUF, the number of 1s and 0s should be equal to

half of the total number of bits. For a 512-bit stream, the ideal number of 0s and 1s is 256 bits each. So, generally, the strongest half of the cells can be 128 bits of strong 0's and 128 bits of strong 1's.

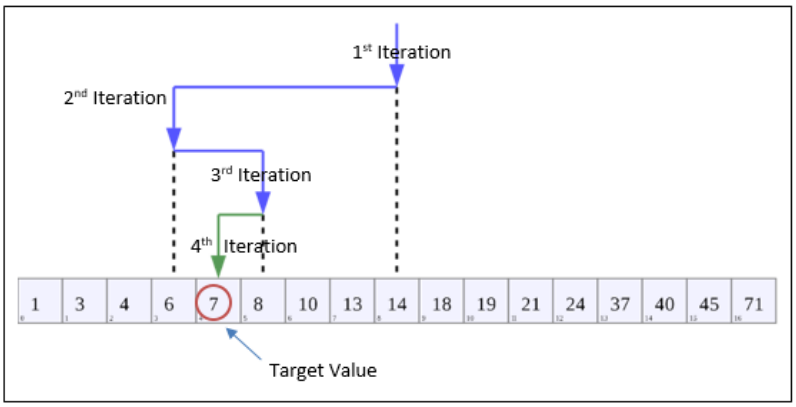

Fig. 4. Illustration of the Working Principle of Binary Search

Using the binary search method, this research investigates the power-off time required to obtain 128 bits of strongest 0s and strongest 1s. At first, the initial value of the time *T* will be set as 1024 (2^{10}), and that of increment/decrement *x* is set at half of *T* at 512. This value of 2^{10} is chosen to avoid using decimal values to delay for repeated halving. After being powered off for a duration of T milliseconds, the flipped cells are counted to find the strongest cells and their optimal power-off time. If the number of flipped bits is less than 128, the value of T will be increased by *x* and vice versa *x* if it is greater than 128. The value of *x* is then halved for the next loop. The loop will be running until the desired amount of 128 flipped bits is found. The algorithm for finding the strongest bits and the optimal power-off time is shown in Figure 5.

The algorithm is then implemented on an Arduino microcontroller to control all the other microcontrollers and SRAM chips under test. To save computation time, a lookup table (LUT) is used for bit counts in individual bytes instead of calculating them for every loop.

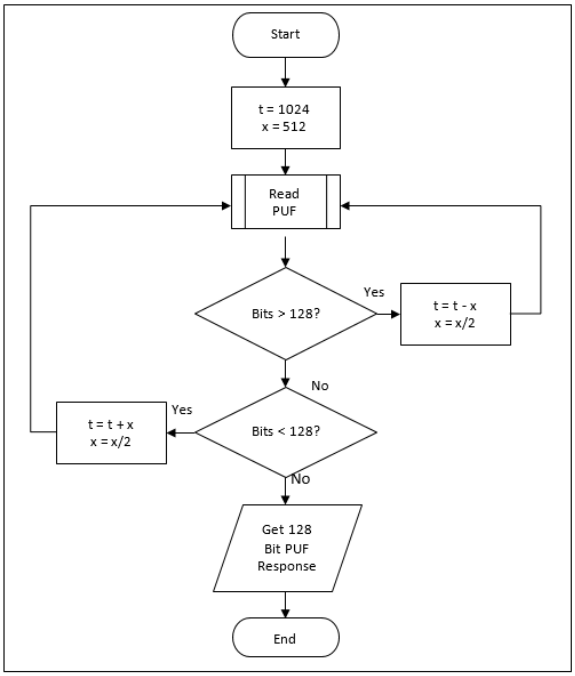

Fig. 5. Algorithm for Finding Strongest Bits and Optimal Power- off Time.

5. Results and Discussion

5.1 Experimental Setup and Results Summary

A total of 17 devices consisting of 12 SRAMs that are embedded in microcontroller chips and 5 off-the-shelf SRAM ICs have been tested for the SRAM PUF characteristics in this research. These chips are selected due to their prominence in many IoT applications on the market and its utilization in many prototype devices. The SRAMs which are from different chips come in different package sizes and capacity. They are also fabricated in different transistor technology. The details of each type of chip are summarized in Table 1. Five AtMega328P microcontrollers from Arduino UNO boards denoted as chips A1 to A5; three ESP8266 microcontrollers from ESP8266 Node MCU denoted as chips B1 to B3; four STM32F103C8 microcontrollers from STM32 Blue-Pill Boards denoted as C1 to C4; five 23LC1024 serial CMOS SRAM chips from Microchip denoted as D1 to D5.

Table 1
Details of Chips Used in the Experiment

Chip Name	Type	Used In	Quantity	Instance labels
AtMega328P	Microcontroller	Arduino UNO	5	A1, A2, A3, A4, A5
ESP8266	Microcontroller	Node MCU	3	B1, B2, B3
STM32F103C8	Microcontroller	Blue Pill Development Board	4	C1, C2, C3, C4
23LC1024	SRAM Chip	N/A	5	D1, D2, D3, D4, D5

The summary of the experimental settings performed in this research is presented in Table 2. Three types of experiments were conducted: (i) finding PUF characteristics using the majority voting method under various temperature settings, (ii) finding PUF characteristics using data remanence method at various power-off time, and (iii) identifying the optimal power-off time for strong PUF characteristics using data remanence method. To test the characteristics under different temperatures, the microcontrollers are placed in air-tight insulating container made of expended polystyrene foam and cooled or heated the whole container from outside.

Table 2
Summary of Experimental Settings

No	Experiment Description		Temperature Range	Iterations
1	SRAM PUF Characteristics using Majority Voting		25°C	1000
			-15°C to 80°C	100
2	Data Remanence	200 ms	25°C	100
		280 ms		100
		300 ms		100
		500 ms		100
3	Data Remanence	at optimal power off time	25°C	20
		at optimal power off time	-15°C to 80°C	20

In the first experiment, PUF characteristics are obtained using majority voting. The characteristics include Bit Error Rate (BER), Biasness, Randomness, Uniqueness, and Stability of all the microcontrollers and SRAM chips tested in this research. The second experiment characterized PUF at room temperature (25°C) at different power-off times. The third experiment used a binary search algorithm to find the optimal power-off time to obtain the strong PUF characteristics at room temperature (25°C) and within the temperature range of -15°C to 80°C, which is the recommended

operating temperature of the chips being used. Table 3 summarizes the SRAM PUF characterizations resulting from the three experiments, detailed in the following sections.

Table 3
Result Summary of SRAM PUF Characterizations

SRAM Type	Method	Majority Voting		Data Remanence (Strong 1 Only)		Data Remanence (Strong 0 Only)		Data Remanence (Strong 1 and Strong 0)	
		25°C	-15°C to 80°C	25°C T = 300 ms	25°C T = 500 ms	25°C T = 300 ms	25°C T = 500 ms	25°C T = 300 ms	25°C $T = T_{optimal}$
AtMega 328P	Error Rate	1.86 %	2.39 %	0.0138%	0.3105%	0.0293%	0.546%	$3.77x10^{-7}$	0 %
	Biasness	61.82 %	62.07 %	38.086%	46.680%	66.992%	65.039%	50 %	50 %
	Uniqueness	48.40 %	40.88 %	N/A	N/A	N/A	N/A	48.40 %	48.40 %
	Stability	95.83 %	85.85 %	99.805%	97.852%	99.707%	97.266%	99.983 %	100 %
ESP8266	Error Rate	1.68 %	2.34 %	0.0245%	0.670%	0.0415%	0.619%	$6.51x10^{-6}$	1.9×10^{-7}
	Biasness	63.01 %	50 %	40.184%	47.501%	62.783%	60.319%	50 %	50 %
	Uniqueness	52.21 %	50.29 %	N/A	N/A	N/A	N/A	52.21 %	52.21 %
	Stability	94.95 %	81.45 %	99.715%	96.981%	99.543%	97.312%	99.975 %	99.999 %
STM32 F103C8	Error Rate	1.63 %	3.34 %	0.0189%	0.504%	0.0448%	0.785%	$2.44x10^{-6}$	0 %
	Biasness	61.10 %	58.89 %	39.254%	49.343%	60.992%	58.352%	50 %	50 %
	Uniqueness	47.98 %	48.37 %	N/A	N/A	N/A	N/A	47.98 %	47.98 %
	Stability	95.15 %	78.53 %	99.248%	96.841%	99.413%	96.166%	99.979 %	100 %
23LC1024	Error Rate	2.46 %	5.46 %	0.0149%	0.324%	0.0304%	0.574%	$7.81x10^{-6}$	9.7×10^{-8}
	Biasness	51.66 %	49.66 %	32.184%	45.425%	55.845%	58.456%	50 %	50 %
	Uniqueness	45.13 %	42.13 %	N/A	N/A	N/A	N/A	45.13 %	45.13 %
	Stability	99.54 %	76.85 %	99.805%	97.918%	98.517%	96.145%	99.963 %	99.999 %

5.2 SRAM PUF Characterization using Majority Voting

PUF streams were collected for multiple iterations from the startup values of SRAMs at predefined memory addresses. The result of the first five iterations of the AtMega328P microcontrollers is presented in Figure 6.

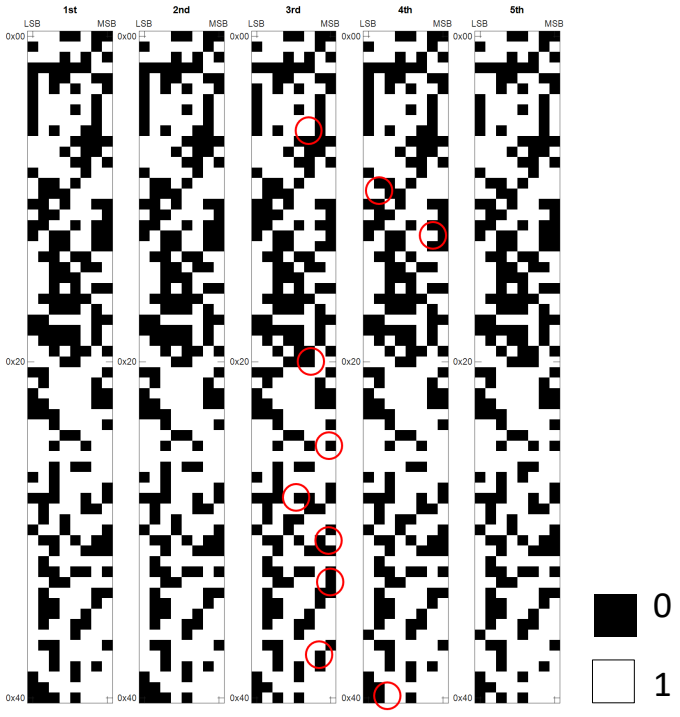

Fig. 6. The first five iterations of initial SRAM values from the AtMega328p microcontroller

The black cell represents bit '*0*' whereas the white cell represents bit '*1'*. Some of the bit flipping captured at some of the iterations due to intrinsic noise nature are denoted by red circles. Only a few unstable bits are flipped while the remaining bits are stable, holding their value of '0's (black cells) or '1's (white cells) all the time. The reference PUF streams are obtained by majority voting for each individual chip after one hundred iterations. The experiment was done in room temperature for one thousand iterations in order to get more precise data and to compute the standard deviation for better analysis.

5.2.1 Bit Error Rate

The first characteristic of SRAM PUF being analyzed is the bit error rate (BER). Figure 7 plots the error rate for the first 100 iterations of an AtMega238 microcontroller from Arduino UNO boards. The error rate ranges from 0.2% to 2.4%, with a standard deviation of around 0.5%. However, the error rate never reaches zero; there are always a few bits flipping for each iteration. Therefore, the stability of 100% could not be achieved by using the majority voting method alone.

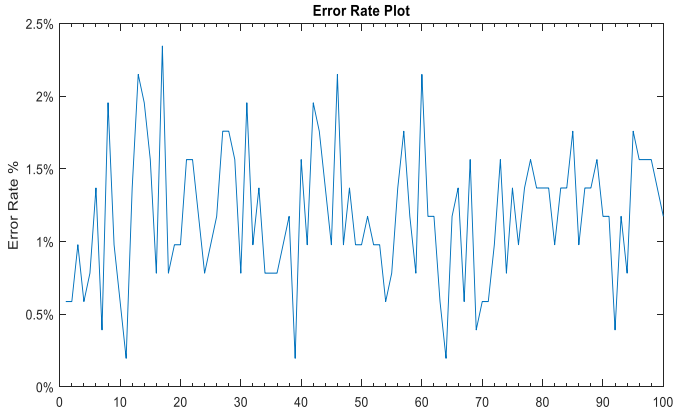

Fig. 7. The plot of BER for 100 iterations.

Similar results of BER were found for other microcontrollers and SRAM chips, despite of their different physical package size and memory capacity. All the evaluation results of BER are summarized in the third column and the fourth column of Table 3. The AtMega328P, STM32F103C8, and ESP8266 microcontroller chips have slightly better performance with relatively lower error rates averaging about 1.7% with standard deviations of about 0.5%, while the serial SRAM chips have an average error rate of 2.5% with standard deviations of about 0.7% at room temperature. For the temperature range of -15^{o}C to 80^{o}C, the error rate of the microcontroller chips is averaging about 2.7%, while the serial SRAM chips have an average error rate of 5.5%. The lower the temperature is, the more effect it has on the error rate of the SRAM PUF. But the temperature above 30^{o}C does not affect the error rate significantly. This is consistent with the property of SRAM PUFs which is affected most by the decrease in temperature [46].

5.2.2 Biasness

As for the biasness, the SRAM PUF of most microcontroller chips is biased towards '1' with average uniformity of slightly more than 60%. They all have a relatively low standard deviation of about 0.5%. Column 3 and column 4 of Table 3 summarize the biasness results for room temperature and temperature range of -15^{o}C to 80^{o}C, respectively. However, serial SRAM chips are well uniformed with an average biasness of around 50% and a standard deviation of 0.9%. Therefore, SRAM PUFs are still within the reasonable range to be used for random number generations.

5.2.3 Randomness

Randomness is the measure of uncertainty or unpredictability of the data. The SRAM PUFs of all the microcontrollers and SRAM chips have more than 90% of randomness. The 23LC1024 serial SRAM chips have the best randomness with approximately 99% consistently. The AtMega328P and ESP8266 microcontrollers are also not very far off, having average randomness of about 95%. One STM32F103C8 microcontroller C1 has the lowest randomness of 91%. In a nutshell, all the microcontrollers and SRAM chips have high randomness of more than 90%, which means they are all suitable to be used as random number generator. A comparison of the randomness of each chip is illustrated in Figure 8.

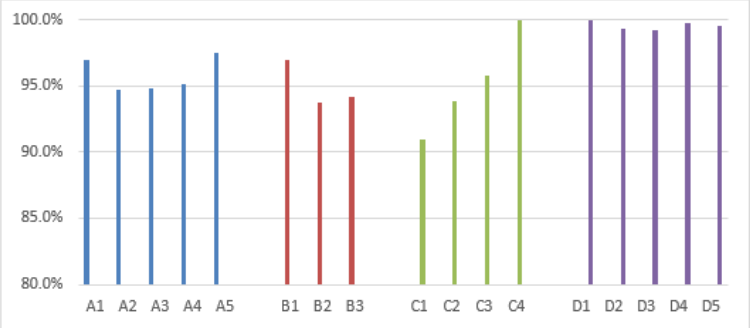

Fig. 8. The randomness of each chip compared.

5.2.4 Uniqueness

Uniqueness, also known as inter-difference, represents how different or similar the reference PUFs of individual chips can be utilized as unique authentication keys. Since PUF deals with binary data, a uniqueness of 50% and the smallest deviation is desirable. As shown in columns 3 and 4 of Table 3, the AtMega328P microcontrollers from Arduino UNO boards have the best uniqueness values with an average uniqueness of 48.40 % and a standard deviation of 2.12%. STM32 microcontrollers come in second with slightly worse results. ESP8266 microcontrollers and serial SRAM ICs are the worst performers in terms of uniqueness, both having a standard deviation higher than 10%.

5.2.5 *Stability*

The stability of SRAM PUF is measured by the number of stable bits; the bits that never changed their values during the one hundred iterations. All the chips have 75% or more stable bits. The STM32F103C8 microcontrollers have the highest number of stable bits, about 90%. Other microcontrollers are not very far off except for the serial SRAM chips, which have the least number of stable bits with about 82% on average. The SRAM chips are also the least consistent in terms of stability during this research. The stability comparison among microcontrollers and the SRAM chip is presented in column 3 and 4 in Table 3.

Both stable bits and non-stable bits, according to their respective memory addresses for each chip, can be identified by majority voting. Although this method can find the least stable bits in SRAM PUF, it cannot determine the strongest or most stable bits among the remaining stable bits. Moreover, the error rate and the stability of the SRAM PUF still need to be improved to be used in authentication applications.

5.3 SRAM PUF Characterization based on Data Remanence

This section demonstrates the improvements in the error rate and SRAM PUF when using the data remanence method. To test the data remanence of SRAMs, the entire SRAM array is written with all 0s first. Then, the SRAM PUF responses were observed, and the bits that flip to 1s under different power-off times were recorded as strong-1 bits. A similar procedure was done for strong-0 bits by writing all the cells with 1s and recording the bits that flipped to 0 after powering off. Different microcontrollers behave differently under the same power-off time, but for most microcontrollers, the cells start flipping around 200ms to 300ms.

SRAM PUF response for both strong-1 and strong-0 cells of an AtMeta328P microcontroller under different power-off times is presented in Figure 9. As shown in the graph, memory cells start flipping their values from around 250ms after power-off. At 280ms, a few of the cells are flipped and considered the strongest cells. Most of the strong-1 and strong-0 cells are flipped at around 300ms. Column 5 in Table 3 shows SRAM PUF characteristics at 300ms using the data remanence method. After 500ms, the cell flipping has stopped, and the PUF response is almost indistinguishable from the response obtained by majority voting, which has a power-off time of several minutes. As shown in Table 3, BER is reduced tremendously to near zero while the stability has increased to nearly 100% under the data remanence method when considering both strong-1 and strong-0 cells.

By utilizing the data remanence, the strongest cells for PUF can be selected. Since they are the strongest cells, they are less likely to make errors and typically much more stable than the weak cells. However, using only the strongest cells will make the biasness and uniqueness of the overall system decline. For the AtMega328P microcontroller, the values are well balanced at the power-off time *T* of around 300ms.

5.4 Optimal Power-off Duration Time for Strong Bits based on Data Remanence

The power-off duration time where all the well-balanced characteristics differ from chip to chip is called optimal power-off time. The optimal power-off time is when half of the ideal number of bits flipped for each of the microcontrollers and SRAM chips. The optimal power-off time is different for each of the microcontroller chips. So, we propose to find the optimal power-off time for each of the microcontroller and SRAM chips using binary search. The values range from about 120ms to 630ms, as shown in Table 4. However, the type of chip does not seem to correlate to the values of the optimal power-off time.

The binary search for optimal power-off time is also done under various environmental temperatures ranging from -8^{o}C to 75^{o}C. Although temperatures above 20^{o}C have no apparent effect on the optimal power-off time, lower temperatures can drastically increase the optimal power-off time. This is because the SRAMs have a higher tendency to undergo data remanence under lower temperatures and can maintain the data longer without any power [46]. However, the error rate and the stability are not affected by lower temperatures for the strong-1 and strong-0 cells, which are selected using the optimal power-off times.

As shown in Table 4, there are many improvements in terms of error rate and stability. For AtMega328 and STM32 microcontrollers, the error rate could be reduced to 0 % by using the optimal power-off time. It is also reduced to 0 % in two out of three of the ESP8266 microcontrollers and four out of five of the 23LC1024 serial SRAM chips respectively.

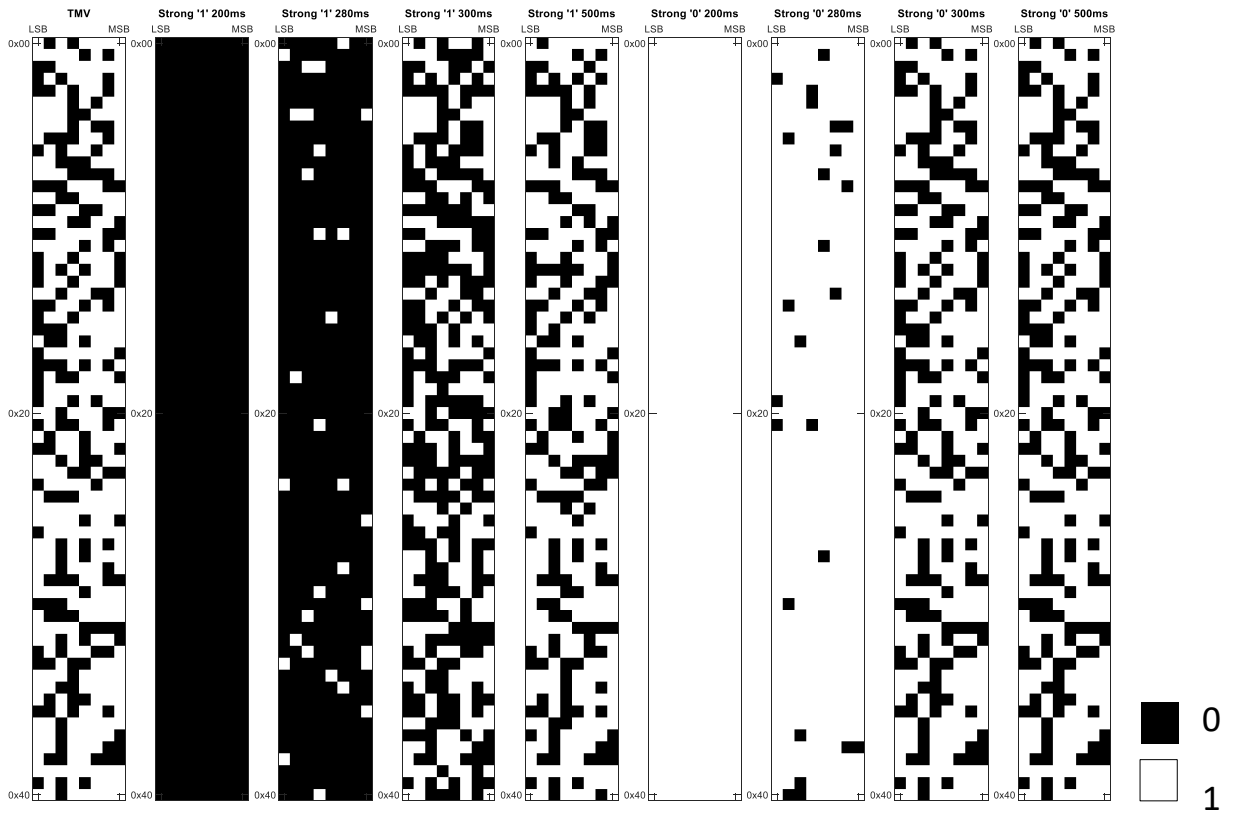

Fig. 9. SRAM PUF Response of AtMega328P microcontroller in multiple different power-off events

Table 4
Optimal Power-off Time (ms) for SRAM on selected microcontrollers

Chip		Optimal Power off Time (ms)		Error Rate $T = T_{Optimal}$	Stability $T = T_{Optimal}$
		Strong 1	Strong 0		
Atmega 328P	A1	293	193	0%	100%
	A2	233	134	0%	100%
	A3	361	209	0%	100%
	A4	628	371	0%	100%
	A5	525	361	0%	100%
ESP8266	B1	352	232	5.7×10^{-7}	99.998%
	B2	270	148	0%	100%
	B3	460	257	0%	100%
STM32 F103C8	C1	250	120	0%	100%
	C2	417	230	0%	100%
	C3	404	248	0%	100%
	C4	389	383	0%	100%
23LC1024	D1	468	452	0%	100%
	D2	316	386	4.85×10^{-7}	99.998%
	D3	286	232	0%	100%
	D4	385	341	0%	100%
	D5	398	338	0%	100%

As for the stability, all the chips improved significantly, reaching the stability of 100 % except one ESP8266 microcontroller and one 23LC1024 serial SRAM chip, having one unstable bit. They still manage to obtain stability of more than 99.999% during the 100 iterations.

The binary search takes significantly fewer iterations to find the optimal power off-time compared to linear search. Thus, the overall computational time has been reduced drastically. For instance, binary search method took 51.57 seconds or nine iterations to find the optimal power-off time of 626ms compared to linear search, with one-millisecond resolution, would take 626 iterations or approximately 33,626 seconds, which is almost one hour, to find the result.

6. Conclusion

The characteristics, namely the bit error rate, biasness, randomness, uniqueness, and stability of several microcontrollers and serial SRAM chips being used in many current IoT applications, have been investigated thoroughly to verify the potential applications of SRAM PUF. The chips used in this research are AtMega328P, ESP8266, STM32F103C8 microcontrollers and 23LC1024 serial SRAM chips. According to the evaluation results, the SRAM PUFs can be utilized as a unique identity of each semiconductor chip for security and authentication purposes. They have very good randomness, which can be used for true random number generation. With very high uniqueness, their PUF responses are practically impossible to be duplicated. So, they are also suitable for identification and device authentication. However, due to the relatively high error rate and non-ideal stability, a binary search algorithm for determining the optimal power off time was proposed in this work to select the stable bit of SRAM PUF more accurately by utilizing the data remanence nature of SRAMs; the strongest 1 and strongest 0 SRAM PUF cells have been identified, which achieved up to 100% stability and 0% error rate on AtMega328P and STM328F103C8 microcontrollers. In the worst case, the ESP8266 microcontrollers could achieve an error rate of 1.9×10^{-7} % and more than 99.999% stability. Moreover, the computational time can significantly be reduced compared to other conventional methods such as linear search. These results showed that the binary search algorithm for finding the optimal power off time could be considered one of the best methods for a stable bit selection process.

Acknowledgment

This work is supported by Universiti Teknologi Malaysia (UTM) Research University Grant with vote number Q.K130000.3843.20J98.

References

[1] Lasse Lueth, Knud. “State of the IoT 2020: 12 Billion IoT Connections, Surpassing Non-IoT for the First Time.” IoT Analytics, November 19, 2020. https://iot-analytics.com/state-of-the-iot-2020-12-billion-iot-connections-surpassing-non-iot-for-the-first-time/.

[2] Mittal, Sachin, Wang Tsz Tam, and Chris Ko. "Internet of Things: The pillar of artificial intelligence." Report produced by Asian Insights Office: DBS Group (2018).

[3] Ashton, Kevin. "That ‘internet of things’ thing." *RFID journal* 22, no. 7 (2009): 97-114.

[4] Aman, Muhammad N., Kee Chaing Chua, and Biplab Sikdar. "Position paper: Physical unclonable functions for iot security." In Proceedings of the 2nd ACM international workshop on IoT privacy, trust, and security, pp. 10-13. 2016. https://doi.org/10.1145/2899007.2899013

[5] Noh, Jaewon, Jeehyeong Kim, Giwon Kwon, and Sunghyun Cho. "Secure key exchange scheme for WPA/WPA2-PSK using public key cryptography." In *2016 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, pp. 1-4. IEEE, 2016. https://doi.org/10.1109/ICCE-Asia.2016.7804782

[6] Thomas, Sylvia. "A detailed review on physical unclonable function circuits for hardware security." In *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pp. 609-612. IEEE, 2018.

[7] Thomas, Sylvia. "A detailed review on physical unclonable function circuits for hardware security." In *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pp. 609-612. IEEE, 2018.
[8] Chatterjee, Urbi, Rajat Subhra Chakraborty, and Debdeep Mukhopadhyay. "A PUF-based secure communication protocol for IoT." *ACM Transactions on Embedded Computing Systems (TECS)* 16, no. 3 (2017): 1-25. https://doi.org/10.1145/3005715
[9] McGrath, Thomas, Ibrahim E. Bagci, Zhiming M. Wang, Utz Roedig, and Robert J. Young. "A puf taxonomy." *Applied physics reviews* 6, no. 1 (2019). https://doi.org/10.1063/1.5079407
[10] Maes, Roel, and Roel Maes. Physically unclonable functions: Concept and constructions. Springer Berlin Heidelberg, 2013. https://doi.org/10.1007/978-3-642-41395-7
[11] Sklavos, Nicolas. "Securing communication devices via physical unclonable functions (PUFs)." In ISSE 2013 Securing Electronic Business Processes: Highlights of the *Information Security Solutions Europe 2013 Conference*, pp. 253-261. Springer Fachmedien Wiesbaden, 2013. https://doi.org/10.1007/978-3-658-03371-2_22
[12] Maes, Roel, and Ingrid Verbauwhede. "Physically unclonable functions: A study on the state of the art and future research directions." *Towards Hardware-Intrinsic Security: Foundations and* Practice (2010): 3-37. https://doi.org/10.1007/978-3-642-14452-3_1
[13] Lipps, Christoph, Andreas Weinand, Dennis Krummacker, Christoph Fischer, and Hans D. Schotten. "Proof of concept for IoT device authentication based on SRAM PUFs using ATMEGA 2560-MCU." In *2018 1st International Conference on Data Intelligence and Security (ICDIS)*, pp. 36-42. IEEE, 2018. https://doi.org/10.1109/ICDIS.2018.00013
[14] Babaei, Armin, and Gregor Schiele. "Physical unclonable functions in the internet of things: State of the art and open challenges." *Sensors* 19, no. 14 (2019): 3208. https://doi.org/10.3390/s19143208
[15] Pehl, Michael, Akshara Ranjit Punnakkal, Matthias Hiller, and Helmut Graeb. "Advanced performance metrics for physical unclonable functions." In *2014 International Symposium on Integrated Circuits (ISIC)*, pp. 136-139. IEEE, 2014. https://doi.org/10.1109/ISICIR.2014.7029527
[16] Fujiwara, Hidehiro, Makoto Yabuuchi, and Koji Nii. "Assessing uniqueness and reliability of SRAM-based Physical Unclonable Functions from silicon measurements in 45-nm bulk CMOS." In *Fifteenth International Symposium on Quality Electronic Design*, pp. 523-528. IEEE, 2014. https://doi.org/10.1109/ISQED.2014.6783371
[17] Holcomb, Daniel E., Wayne P. Burleson, and Kevin Fu. "Initial SRAM state as a fingerprint and source of true random numbers for RFID tags." In *Proceedings of the Conference on RFID Security*, vol. 7, no. 2, p. 01. 2007.
[18] Cortez, Mafalda, Said Hamdioui, and Ryoichi Ishihara. "Design dependent SRAM PUF robustness analysis." In 2015 *16th Latin-American Test Symposium (LATS)*, pp. 1-6. IEEE, 2015. https://doi.org/10.1109/LATW.2015.7102498
[19] Cortez, Mafalda, Apurva Dargar, Said Hamdioui, and Geert-Jan Schrijen. "Modeling SRAM start-up behavior for physical unclonable functions." In *2012 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, pp. 1-6. IEEE, 2012. https://doi.org/10.1109/DFT.2012.6378190
[20] Zhang, Shen, Bin Gao, Dong Wu, Huaqiang Wu, and He Qian. "Evaluation and optimization of physical unclonable function (PUF) based on the variability of FinFET SRAM." In *2017 International Conference on Electron Devices and Solid-State Circuits (EDSSC)*, pp. 1-2. IEEE, 2017. https://doi.org/10.1109/EDSSC.2017.8126474
[21] Narasimham, Balaji, Dan Reed, Saket Gupta, Ennis T. Ogawa, Yifei Zhang, and J. K. Wang. "SRAM PUF quality and reliability comparison for 28 nm planar vs. 16 nm FinFET CMOS processes." In *2017 IEEE International Reliability Physics Symposium (IRPS)*, pp. PM-11. IEEE, 2017. https://doi.org/10.1109/IRPS.2017.7936393
[22] Wang, Wendong, Adit Singh, Ujjwal Guin, and Abhijit Chatterjee. "Exploiting power supply ramp rate for calibrating cell strength in SRAM PUFs." In *2018 IEEE 19th Latin-American Test Symposium (LATS)*, pp. 1-6. IEEE, 2018. https://doi.org/10.1109/LATW.2018.8349685
[23] Cui, Ziyang, Baikun Zheng, Yanhao Piao, Shiyu Liu, Ronghao Xie, and Hirofumi Shinohara. "Measurement of mismatch factor and noise of SRAM PUF using small bias voltage." In *2017 International Conference of Microelectronic Test Structures (ICMTS)*, pp. 1-4. IEEE, 2017.
[24] Elshafiey, Abdelrahman T., Payman Zarkesh-Ha, and Joshua Trujillo. "The effect of power supply ramp time on SRAM PUFs." In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 946-949. IEEE, 2017. https://doi.org/10.1109/MWSCAS.2017.8053081
[25] Van Aubel, Pol, Daniel J. Bernstein, and Ruben Niederhagen. "Investigating SRAM pufs in large cpus and gpus." In Security, Privacy, and Applied Cryptography Engineering: 5th International Conference, SPACE 2015, Jaipur, India, October 3-7, 2015, Proceedings 5, pp. 228-247. *Springer International Publishing*, 2015. https://doi.org/10.1007/978-3-319-24126-5_14
[26] Wilde, Florian. "Large scale characterization of SRAM on Infineon XMC microcontrollers as PUF." In *Proceedings of the Fourth Workshop on Cryptography and Security in Computing Systems*, pp. 13-18. 2017. https://doi.org/10.1145/3031836.3031839

[27] Takeuchi, Kiyoshi, Tomoko Mizutani, Takuya Saraya, Masaharu Kobayashi, Toshiro Hiramoto, and Hirofumi Shinohara. "Measurement of SRAM power-up state for PUF applications using an addressable SRAM cell array test structure." In *2016 International Conference on Microelectronic Test Structures (ICMTS)*, pp. 130-134. IEEE, 2016. https://doi.org/10.1109/ICMTS.2016.7476191

[28] Takeuchi, Kiyoshi, Tomoko Mizutani, Hirofumi Shinohara, Takuya Saraya, Masaharu Kobayashi, and Toshiro Hiramoto. "Measurement of static random access memory power-up state using an addressable cell array test structure." *IEEE Transactions on Semiconductor Manufacturing* 30, no. 3 (2017): 209-215. https://doi.org/10.1109/TSM.2017.2692805

[29] Mispan, Mohd Syafiq, Basel Halak, and Mark Zwolinski. "NBTI aging evaluation of PUF-based differential architectures." In *2016 IEEE 22nd International Symposium on On-Line Testing and Robust System Design (IOLTS)*, pp. 103-108. IEEE, 2016. https://doi.org/10.1109/IOLTS.2016.7604680

[30] Trujillo, Joshua, Christian Merino, and Payman Zarkesh-Ha. "SRAM physically unclonable functions implemented on silicon germanium." In *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1-4. IEEE, 2019. https://doi.org/10.1109/ISCAS.2019.8702437

[31] Zhang, Le, Chip-Hong Chang, Zhi Hui Kong, and Chao Qun Liu. "Statistical analysis and design of 6T SRAM cell for physical unclonable function with dual application modes." In *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1410-1413. IEEE, 2015. https://doi.org/10.1109/ISCAS.2015.7168907

[32] Barbareschi, Mario, Ermanno Battista, Antonino Mazzeo, and Nicola Mazzocca. "Testing 90 nm microcontroller SRAM PUF quality." In *2015 10th International Conference on Design & Technology of Integrated Systems in Nanoscale Era (DTIS)*, pp. 1-6. IEEE, 2015. https://doi.org/10.1109/DTIS.2015.7127360

[33] Liao, Zhonghao, and Yong Guan. "The cell dependency analysis on learning sram power-up states." In *2018 Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*, pp. 37-43. IEEE, 2018. https://doi.org/10.1109/AsianHOST.2018.8607167

[34] Liao, Zhonghao, George T. Amariucai, Raymond KW Wong, and Yong Guan. "The impact of discharge inversion effect on learning sram power-up statistics." In *2017 Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*, pp. 31-36. IEEE, 2017. https://doi.org/10.1109/AsianHOST.2017.8353991

[35] Alheyasat, A., Gabriel Torrens, S. Bota, and Bartomeu Alorda. "Weak and Strong SRAM cells analysis in embedded memories for PUF applications." In *2019 XXXIV Conference on Design of Circuits and Integrated Systems (DCIS)*, pp. 1-6. IEEE, 2019. https://doi.org/10.1109/DCIS201949030.2019.8959939

[36] Handschuh, Helena. "Hardware intrinsic security based on SRAM PUFs: Tales from the industry." In *2011 IEEE International Symposium on Hardware-Oriented Security and Trust*, pp. 127-127. IEEE, 2011. https://doi.org/10.1109/HST.2011.5955009

[37] Skorobogatov, Sergei. Low temperature data remanence in static RAM. No. UCAM-CL-TR-536. University of Cambridge, Computer Laboratory, 2002.

[38] Liu, Muqing, Chen Zhou, Qianying Tang, Keshab K. Parhi, and Chris H. Kim. "A data remanence-based approach to generate 100% stable keys from an SRAM physical unclonable function." In *2017 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 1-6. IEEE, 2017. https://doi.org/10.1109/ISLPED.2017.8009192

[39] Setyawan Sajim, Ade. "Open-source software-based sram-puf for secure data and key storage using off-the-shelf sram." (2018).

[40] Aung, Pyi Phyo, Koichiro Mashiko, Nordinah Binti Ismail, and Ooi Chia Yee. "Evaluation of SRAM PUF characteristics and generation of stable bits for IoT security." In *Emerging Trends in Intelligent Computing and Informatics: Data Science, Intelligent Information Systems and Smart Computing* 4, pp. 441-450. Springer International Publishing, 2020. https://doi.org/10.1007/978-3-030-33582-3_42

[41] Böhm, Christoph, Maximilian Hofer, and Wolfgang Pribyl. "A microcontroller sram-puf." In *2011 5th International Conference on Network and System Security*, pp. 269-273. IEEE, 2011. https://doi.org/10.1109/ICNSS.2011.6060013

[42] Deutschmann, Martin, Sandra-Lisa Lattacher LejlaIriskic, and Felix Stornig Mario Münzer. "Research on the Applications of Physically Unclonable Functions within the Internet of Things" (2018).

[43] Van Den Berg, Robbert. "Entropy analysis of physical unclonable functions." Ph. D. dissertation, Department of Mathematics and Computer Science, Eindhoven University, Eindhoven (2012).

[44] Schaub, Alexander, Olivier Rioul, and Joseph J. Boutros. "Entropy estimation of physically unclonable functions via Chow parameters." In *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 698-704. IEEE, 2019. https://doi.org/10.1109/ALLERTON.2019.8919927

[45] Wu, Ling, Sheng Liu, Baoling Zhao, Weinan Wu, and Baozhong Zhu. "The research of the application of the binary search algorithm of RFID system in the supermarket shopping information identification." *EURASIP journal on wireless communications and networking 2019 (2019)*: 1-10. https://doi.org/10.1186/s13638-019-1343-2

[46] Anagnostopoulos, Nikolaos Athanasios, Tolga Arul, Markus Rosenstihl, André Schaller, Sebastian Gabmeyer, and Stefan Katzenbeisser. "Attacking SRAM PUFs using very-low-temperature data remanence." *Microprocessors and Microsystems* 71 (2019): 102864. https://doi.org/10.1016/j.micpro.2019.102864