Evolutionary Cipher against Differential Power Attack

Tang ming[1,2], Meng Qinshu[2], Zhang Huanguo[1,2], Gao Si[2], Dou Qin[2], Shen Fei[2], Li Du[2]

1(State Key Lab. of AIS&TC, Ministry of Education, Wuhan University in China, m.tang@126.com)

2(School of Computer, Wuhan University, Wuhan 430072, China)

Abstract: DPA attack is one of most threatening SCA attacks, this paper focuses on research of DPA resistance. There are two phases in DPA attacks: collection and analyzing which can be utilized to construct different countermeasures against DPAs, such as balancing technologies aim at analyzing. We propose a new idea with dynamic structure algorithm to resist DPAs and call this measure as evolutionary cipher which can effectively resist DPA attacks based on destroying differential power computation model proposed by kocher. Moreover, evolutionary cipher opens up a new idea to design safety cryptographic algorithm for it can resist both DPA attack and some mathematic attacks as well. Designing principles of evolutionary cipher can be referenced by other dynamic cryptographic algorithms. This paper has theoretically and practically proofed security and effectiveness of evolutionary cipher to resist against DPAs.

Keywords: evolutionary cipher, differential power attack, DPA resistance, dynamic countermeasure

1. Introduction

Security on communication, computation, and storage has been concerned for a long time. Cryptographic algorithms, including symmetric ciphers, public-key ciphers, and hash functions, form a set of primitives that can be used as building blocks to construct security mechanisms that target specific objectives[1]. Cryptosystem designers frequently assume that secrets will be manipulated in closed, reliable computing environments[2]. Based on this “ separation of concerns”, many theory analysis have been well studied, such as differential analysis[3], linear analysis[4].

However, in practice, even if one cryptographic algorithm has been proved to be resistant against all present existed theory analyses methods, security mechanisms based on this cryptographic algorithm are not proved to be safety or reliable. Because this kind of mechanisms may be vulnerable to side channel attacks(SCAs).

.

In cryptography, a side channel attack is any attack based on information gained from the physical implementation of a cryptosystem, rather than brute force or theoretical weaknesses in the algorithms (compare cryptanalysis) [5].

Over the passed 10 years, there existed some effective SCA attacks such as Power attacks(combined with SPA, DPA, High-Order DPAs), Fault attacks(Spike, Glitch, Optical, Electromagnetic), Timing attacks, and Electromagnetic analysis.

This paper focuses on countermeasures against DPA attacks which is one of the most threatening SCA attacks. Power analysis attacks are rely on the correlation of power concerned by attackers and corresponding operations or data. Attackers can find some key data during analyzing power samples and making some statistical computations. Comparing with SPAs(simple power attacks) which are more depend on the power collected and understanding the details of cipher operations implementation, DPA attacks are more effective and much harder to resistance with statistical analysis and error correction technology.

Considering of DPA attacks, some countermeasures have been proposed against DPAs. These resistances set difficulties and oracles to DPA attacks from different aspects and can be classified as: masking methods, balancing, and random timing. Using randomly selected masking code on key operations, masking methods[6] can cover the difference between different operations. Based on DPA Leakage Models to quantitatively evaluate leakage info and design balance circuit, balancing[7] can expect to ensure balanced or almost balanced power leakage. And random timing technology[8] and disturbing clock frequency[9] can improve uncertainty of certain power sample by attackers. Unfortunately, these existing countermeasures are all have defects in varying degrees, and we'll carefully introduce and analyze these 3 kinds of methods in section 3.

This paper proposes a new method to resist DPAs based on the power analyzing model built by kocher[2]. Our method combined intelligent computing and cipher design policies to ensure both dynamic structure and more security of cryptographic algorithm. We call this designing method as evolutionary cipher, and we have made some progress on automatic design SBOX[10,11], bent function[12], hash functions[13] with evolutionary cipher. The purposes of all these results are

improving the security of cryptographic algorithm to resist mathematic analysis.

The reason why Evolutionary cipher can resist DPA attacks is its dynamic unpredictable structure which can destroy the differential power analyzing model proposed by kocher[2], so that prevents attackers to identify correct data even if they collect enough power cycles and filter random disturbing elements. Section 4 and 5 illustrate and prove security and effectiveness of evolutionary cipher against DPA attacks respectively.

The organization of this paper is: section 2 carefully introduces principle of DPA attacks, which is composed of two phases; section 3 analyzes three different existing countermeasures against DPAs, which are all hot methods till now; section 4 elaborately introduces how evolutionary cipher can resist DPA attacks from designing principles, resistance against DPAs, and security proof; section 5 makes expensive experiments to verify effectiveness of evolutionary cipher against DPAs; finally last section makes conclusion and future work.

2. Introduction of DPA

Over the past ten years, there have existed some new DPA attacks, but main idea and principles of most DPA attacks[14] are originated from kocher[2]. DPA attacks are composed of 2 phases: the first is power samples collection, and the second is statistical analyzing.

1) power collection: the character of this phase is timing property, which means the power samples record the variety of power during the circuit working. If power cycles are drawn in two dimension, horizontal axis and vertical axis are timing and power respectively.

2) statistical analyzing: based on correlation between variety of power and timing, and some certain observation points may be related with key, attackers could infer some or whole key through power cycle and differential power analyzing model.

Let's introduce the process and principles of differential power attack proposed by kocher[2].

1) Building a selection function $D(C_i, b, k_s)$ which is defined as computing the value of bit b(0 <=b < 32) of the DES intermediate L at the beginning of the 16th round for ciphertext C, where the 6 key bits entering the S box corresponding to bit b are represented by $0 \le k_s < 2^6$ [2].

2) Attackers could observe m encrypting operations and capture two kinds of infomation as followed:

First: C_i represents cipher text which is corresponding to one power trace;

Second: k samples are collection power points and each sample is related to certain time point.

3) Statistically analyzing to get k_s:

First: to get the value of $T_i[j]$ for certain power sample, i is the i[th] power sample and j respects the j[th] sample point;

Second: to compute the value of differential power based on function 1, and only C_i and $T_i[j]$ are variable.

$$\Delta D[j] = \frac{\sum_{i=1}^{m} D(C_i, b, K_s) T_i[j]}{\sum_{i=1}^{m} D(C_i, b, K_s)} - \frac{\sum_{i=1}^{m} (1 - D(C_i, b, K_s)) T_i[j]}{\sum_{i=1}^{m} (1 - D(C_i, b, K_s))}$$
$$\approx 2\left(\frac{\sum_{i=1}^{m} D(C_i, b, K_s) T_i[j]}{\sum_{i=1}^{m} D(C_i, b, K_s)} - \frac{\sum_{i=1}^{m} T_i[j]}{m}\right) \quad (1)$$

Third: If k_s is incorrect, $\lim_{m\to\infty} \Delta D[j] \approx 0$; adversely if k_s is correct, the computed value of $\Delta D[j]$ will not be zero and show spikes in regions where D is correlated to the values being processed. This conclusion has been proofed by kocher[2].

The 2 phases of DPAs can be used to build some kinds of countermeasures against DPA attacks. And we find that there is timing characteristic of power samples collection; however, analyzing phase is more dependent on selection function D.

3. Existing Countermeasures for DPA

3.1 Masking

The masking technique is the most widely used countermeasure against power analysis and timing attacks at a software level and also the most powerful software countermeasure against side channel attacks[15,16].

1) idea： the operations and the key are masked with some random masks at the beginning of computations, and thereafter everything is almost as usual[17]. Using masking resistance, attacker may make mistakes in division and therefore the results of differential power values are not highly

related to correct key.

2）evaluation：masking technology has been widely used in resistance against side channel attacks. Unfortunately, there are some distinct disadvantages: firstly, attackers can filter random disturbing info, unless each bit is split into k shares using any scheme which has the required stochastic properties [18]; secondly, masking often need a very large memory space, and this will make key path larger than usual implementations, so that designers and users often have to balance among the designing cost, run speed and security; thirdly, even designers have considered the size of storage, such as the implementation in [19] to change SBOX design from look up table to logic circuits, there still existed the shortage because that it is impossible to disturb all “zero-value” during multiplication in finite field[20].

3.2 Balancing

Basing on complementary logics such WDDL-AND, Masked-AND[21,22] etc, balancing technologies have been proposed tried to keep the power of whole circuits at a static station to prevent DPA attacks.

Power is the key information which could be captured by attackers. If designers can keep system power in a static level, balancing maybe the most absolute countermeasures against DPA attacks. Until now, there are some technologies based on balancing idea, such as Dual-rail Logic[23] and complementary logics[24].

Quantitative evaluation of leakage model is the precondition for balancing technologies. From the results of these leakage models[25], even some countermeasures such as WDDL complementary logics which have shown almost static power in leakage model experiment results, these countermeasures will be destroyed by improving the concerning precision.

When balancing resistances are used in a large circuit or system, these technologies should be constrained by EDA software kit, which means balancing countermeasures have to use standard circuit designing base and developing language, however, these conditions have not been totally satisfied. Moreover, balancing countermeasures will definitely increase hardware resources at least one time.

3.3 Random Timing

1) purposes:

Random timing belongs to randomizing countermeasures and is one of the most popular countermeasures against DPA attacks. Designers randomly inject timing jitters[8] or change clock frequency[9] with safe true random number generator(TRNG) to disturb concerning points of attackers. Because attacker need each power sample point to be corresponded with certain operation, it is necessary to acquire characters of differential power. Randomly disturbing time is trying to destroy this condition, and then prevent attackers to distinguish each sample of power and corresponding operation. In fact, both random timing and changing clock frequency technologies hope to increase complexity of DPA attacks.

2) Evaluation：

We conclude the characters of random timing technologies as followed:

Advantages: this countermeasure has simple designing idea, very convenient to implementation, and limited hardware resource increase. It also induced dynamic FPGA chip reconfiguration technology[26] into chip designing which can make circuits reconfiguring when they working[8].

Disadvantages: however, there may exist some disadvantages in random timing resistances.

First: this kind of countermeasures only rely on working time of circuits. It is worth mentioning that timing is one character of power concerning process, which means that attacker can verify each round or each operation in circuits of cryptographic algorithms through collecting enough power samples. For example, hardware implementation of block cipher(such as DES[2]), attackers can not only fixed each round but each operation in each round through power samples. So, attacker can confirm the locations of each random timing jitter which is the base of random timing countermeasures.

Second: even if attacker can not find the real locations of random timing jitter(as researchers stated in [8]), we simply analyzed the influence of jitter location for each round of block cipher. Table 1 shows the success ratio against DPA attacks of each round.

Table 1: random timing jitter[8] success ratio against DPA of each round

m / t	1	2	3	4	5	6	7	8	9	10

1	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1	0
2	0.8	0.6222	0.4667	0.3333	0.2222	0.1333	0.0667	0.0222	0	0
3	0.7	0.4667	0.2917	0.1667	0.0833	0.0333	0.0083	0	0	0
4	0.6	0.3333	0.1667	0.0714	0.0238	0.0046	0	0	0	0
5	0.5	0.2222	0.0833	0.0238	0.0039	0	0	0	0	0
6	0.4	0.1333	0.0333	0.0048	0	0	0	0	0	0
7	0.3	0.0667	0.0083	0	0	0	0	0	0	0
8	0.2	0.02	0	0	0	0	0	0	0	0

		2 2								
9	0 . 1	0	0	0	0	0	0	0	0	0
1 0	0	0	0	0	0	0	0	0	0	0

Illumination:

- m is the number of random timing jitters;
- t means the number of concerning points. In AES algorithm, "t=10" means each round corresponds a sample point;
- p_{ij} means success probability against DPA attack at the j^{th} sample point after injecting the i^{th} timing jitter.

For example:

Supposing m=i, and the condition of DPA attack in the j^{th} sample point is i timing jitters are all injected after the j^{th} point, so that timing jitter can not impact the j^{th} point operation time. We suppose only if jitter injected before one round, then the power sample of this round will be disturbed and DPA attack will succeed. And we don't consider other situations such as inject the same number of jitters in each round. Then the probability of DPA attacks is shown as followed:

$$P_{ij} = C_{n-j}^{i} / C_{n}^{i} \quad (i=1\cdots m, j=1\cdots t) \qquad (2)$$

The results in table 1 are the most common situation, and the success ratio of DPA can be increased when attackers make deeper analyzing.

Third: based on our experiments of AES algorithm, injecting timing jitter(or register in hardware implementation) in some round and operations may bring the variety of pre-round to post-round's registers which are injected as timing jitters, and this may magnify the power in these registers. Because this can increase the reverse ratio of registers. If attackers can concern register power(some EDA tools provide this function[27,28], timing jitter may provide more opportunities to DPA attacks. So, randomly injecting timing jitter can not effectively resist against DPA attacks. Fortunately, this measure[8] is still an efficient countermeasure against CPA(Correlation Power Analysis) which need to concern power at certain sample point.

Random timing technology give us inspiration that dynamic circuit may increase complexity of DPA

analyzing or other SCA attacks. In session 4, we will give a new dynamic cryptographic algorithm countermeasure against DPA attacks.

4. Evolutionary Cipher against DPA

After analyzing the existing countermeasures against DPAs, we find there are a few resistances using dynamic circuits to prevent DPA attacks. For examples: masking technologies randomly select masking code, which use RNG(random number generator) or TRNG(true random number generator) to generate random sequence; random timing countermeasures also use RNG(or TRNG) to randomly change clock frequency or inject timing jitters. Besides these dynamic designing measures, now some chip vendors have provided dynamic reconfiguration technologies to configure circuits online[26].

Existing dynamic designing measures have tried to disturb DPA analyzing phase from different aspects. However, these dynamic disturbing countermeasures only have increased the complexity of DPA analyzing. We try to propose a new dynamic design measure which has the ability to destroy the base of DPA analyzing. It is worth to notice that our proposal also can effective resist several mathematic analyzes and has a better scalability as usual countermeasures.

4.1 Designing Principles of Evolutionary Cipher

This paper tries to propose an effective countermeasure against DPAs, in fact, the purpose of evolutionary cipher is to resist different kinds of attacks of cryptographic attacks which include some SCA attacks and mathematic analyzing(off line attacks). We focus on block cipher designing and choose AES algorithm as original algorithm to evolve.

There are some characters of evolutionary cipher as following.

First: dynamic structure of cryptographic algorithm and structure;

Second: Higher levels of security. The purpose of evolutionary cipher is to design safe cryptographic algorithm and resist against different attacks;

Third: more efficient than random or robust selecting;

Forth: complying with cryptographic algorithm designing principles.

In order to keep these characters, evolutionary cipher has the common structure as figure 1.

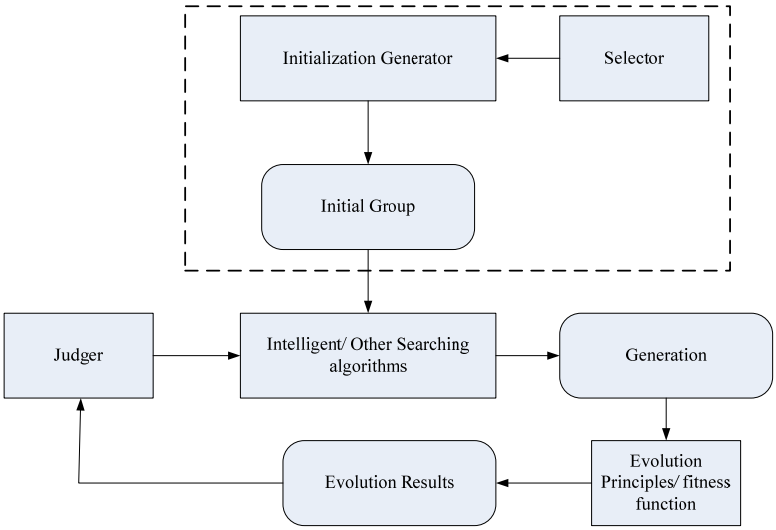


Figure 1: common structure of evolutionary cipher

Illumination:

- selector and judger controls the changingdirection of evolutionary cipher. And both of them are controlled by safe TRNG.
- Fitness functions are evaluation principles of security, and test each generation of evolutionary cipher.
- Evolutionary cipher combines intelligent algorithm and block cipher principles to efficiently and effectively design safe cryptographic algorithm. And these can be implemented in FPGA chips with dynamic reconfiguration technologies[26].

The details and effectiveness against mathematic analyzes of evolutionary cipher have been published in several paper[10,11] and Patent[29]. We conclude the flow of evolutionary cipher as following.

Step1: initialization concludes initial group building, selecting principles, judgment and parameter configuration.

Step2: intelligent search next generation is constrained by fitness function.

Step3: evaluating the new generation and this is the next key point after intelligent searching. Because there are several security evaluations, we should balance among those principles, which equals to solve multi-purpose problems.

Step4: if the condition of output is not satisfied, process will go back to step1 or step2 based on

designing principles. This condition may be constrained by security evaluations or execution time.

Step5: if output condition is satisfied, new generation replaces present one, and dynamic reconfiguration technology can be used to implement evolutionary cipher in chips.

We choose SBOX evolutionary cipher of AES algorithm to introduce how evolutionary cipher resists against DPA attacks in 4.2.

4.2 Evolutionary Cipher against DPAs

4.2.1 Basic Idea

We introduce evolutionary cipher which changes algorithm and logic implementation to destroy the conditions of DPA from collection and analyzing phases. In collection, dynamic implementation induces disturbing value of power; in analysis, evolutionary cipher destroy the differential power computing model proposed by kocher[2].

In order to explain evolutionary cipher's resistance against DPAs, let's recall kocher's differential power computing model in session 2(researchers in Motorola proposed an other model[2] which has the same principle as kocher).

Supposing: A is cryptographic algorithm, D is the function based on kocher's model to compute the b^{th} input of last round in A.

When A has been changed, D will be changed as well. The key problem is whether attacker could find the details of changed D? And this problem equals whether attackers could find the details of changed A?

We classified these problems into two situations. First, it is still public structure of changed A, which means attackers can get every detail of changing results; second, attackers could find that A has been changed(for example, same plaintexts get different ciphertexts), but the specific structure of changed A is secret.

4.2.2 Secrecy of Evolutionary Cipher

The first situation in last part may be regarded as more fitted with public designing principle of cryptographic algorithm. However, after deeply analyzing, we find it is reasonable for evolutionary

cipher and some other dynamic algorithm to keep secret of their changing or changed results. There are two types of operations in evolutionary cipher, and we call them static and dynamic structure parts respectively. In figure 1, static parts include intelligent generator, initial group generator and fitness functions; dynamic parts include selector and judger, which are controlled by TRNG.

Because we use TRNG to control dynamic structure of evolutionary cipher, the next generation can not be participated or controlled by designers, users or attackers, and this point has been coincided with other dynamic design such as random timing[8]. Moreover, we make every detail of evolutionary cipher public to ensure no backdoor in algorithm or implementation. In our opinion, public designing principles of cryptographic algorithm is trying to stop secret in designing algorithm, but the secrecy of evolutionary cipher is secrecy of designing results. This secrecy can also ensure real variety and non-intervention.

So, attackers or even designers of evolutionary cipher can not get knowledge of any future generation, and then can not build correct function of D. Unknown D, it is impossible for attackers to make an effective DPA attacks which are based on differential computation model proposed by kocher.

The reason why evolutionary cipher can resist DPA attacks is attackers can not build correct function of D. Conversely, if attackers could construct correct D function after evolution, evolutionary cipher can not resist against DPAs. We will analyze the probability of losing resistance of this condition at 4.3.

4.3 Security of Evolutionary Cipher

In order to explain security of evolutionary cipher, we illustrate it from several aspects.

4.3.1 When attackers only know the details of evolutionary cipher, whether they can infer certain generation of algorithm?

Because evolutionary cipher is controlled by TRNG, supposing that SBOX is changed by evolutionary cipher[29], each changed value will be selected with same probability. For example, SBOX in AES has the same ratio of $1/256!$ for each candidate (considering inverse operation). It is

impossible for attackers to successfully analyze parameters of SBOX within an effective period of time. So we consider that this situation will be safe to resist DPA attacks.

4.3.2 When attackers could get certain generation of evolutionary cipher, whether they can infer pre or post generation of algorithms?

In this situation, attackers can get certain point of evolutionary cipher algorithm, for example, initial algorithm or original AES algorithm. We borrow the concepts of “forward security and behind security ” in digital signature to illuminate the safety problem existed in this situation.
Based on the common structure of evolutionary cipher in figure 1, dynamic core of evolutionary cipher is TRNG, and this is coincided with the idea of cryptographic algorithm security based on key proposed by Shannon[30]. The difference is both key and algorithm are dynamic in evolutionary cipher.

We use following model to describe an evolutionary cipher system.

$$SF_{gi} = (FG_{gi}, trng, ef) \quad (3)$$

SF_{gi} - means the gith generation of evolutionary cipher;

FG_{gi} - means the initial group for the gith generation;

$trng$ - means random sequence generated by TRNG;

ef - concludes all evolutionary design principles

Among these components in function (3), only FG_{gi} is related with time/generation, and others are all controlled by TRNG. Therefore, security of FG_{gi} is the key problem which is need to analyze.

We analyze security of evolutionary cipher from 2 aspects: first, the length of random sequence influences the security of generation in evolutionary cipher; second, forward and behind security of initial group.

4.3.2.1 Length of Random Sequence

TRNG plays an important role in evolutionary cipher, and there are 2 properties of security TRNG,

first, non-participated with true seed; second, impossible statistical analysis with safe random network.

However, limited length of random sequence will give chance to attackers using robust search to find details of generation. In fact, there are two questions need to solve. One, how long is the safe for random sequence of evolutionary cipher? The other, whether has it a safe length of random sequence for existing evolutionary cipher?

1) safety length of random sequence

Basing on the theory of information proposed by Shannon[30], longer key brings more security of cryptographic algorithm. In theory, one key for one cipher is real safe situation. Therefore, if there is not limitation for the length of random sequence, it is the real safe situation as well. Unfortunately, constrained by present operation and storage resource limitation, existing evolutionary cipher can not use unlimitedly long random sequence.

Standing on the attacker's point of view, it may be easier to find safe length of random sequence. It is supposed that key is static during DPA attacks, and if key changes attackers will make a new attack from scratch. So, we can make such hypothesis that it is safe when random sequence does not same unless key changed. For the longest cycle of key is complied with the length of key, the length of random sequence must not be smaller than that of key. Such as AES algorithm with 128bits key, the length of random sequence in evolutionary cipher will not be less than 128bits.

In conclusion, for security of evolutionary cipher, the length of random sequence will not be smaller than that of key in original cryptographic algorithm.

2）length of random sequence in existing evolutionary cipher

After knowing the safe length of random sequence, we analyze whether did the design of evolutionary SBOX reach safe level at point of random sequence? Such as SBOX in AES algorithm, we have designed evolutionary SBOX in a number of 256!($256! > 2^{128}$) which means SBOX evolutionary design[29] is safe with long enough random sequence. In order to reach safe level, it should choose effective evolutionary objects which have rich enough candidates to reach safe long random sequence.

4.3.2.2 Forward and Behind Security of Different FG_{gi}

As initial group of evolutionary cipher, FG_{gi} is generated by initial group generator. Certain FG_{gi}

may be a new one, or partly selects from FG_{gi-1}. So, there maybe exist some relationship between neighboring FG_{gi} and FG_{gi+1}. This kind of relationship may create association among generations of evolutionary cipher, therefore, forward and behind security of FG_{gi} is the key point of security analysis.

We use the theory of cipher text independence on plaintext to quantitatively analyze the relationship between neighboring FG_{gi}. SF_{gi} and SF_{gi+1} are neighboring evolutionary cipher algorithms and are regarded as plaintext and cipher text respectively. When cipher text is independent on plaintext, it is true that plaintext is independent on cipher text.

Firstly, we introduce the test of independence between plaintext and cipher text. Supposing the size of one block in block cipher is n bits, randomly selects F blocks of plain text $P, P_1, \cdots, P_{F-1}$ and one key. Under the mode of ECB, encrypts plaintexts with the key and outputs F blocks of cipher text $C_0, C_1, \cdots, C_{F-1}$, and computes distances between plaintext and cipher text: $D_i = W(P_i \oplus C_i), 0 \le i \le F-1$ where W means hamming weight between plaintext and cipher text. That number of $D_i (0 \le i \le F-1)$ equals $w (0 \le w \le n)$ is called H_w.

For a safe block cipher, H_w should be satisfied with the distribution of $B(n,1/2)$. Using pearson χ^2 to verify whether H_w is satisfied with $B(n,1/2)$. The expectation of H_w is $E_w = C_n^w \times F / 2^n$, then $\chi^2 = \sum_{i=0}^{n} \frac{(H_i - E_i)^2}{E_i}$. To comparing the Threshold when pearson fitting is χ^2, Significance level is α and freedom is n, if $\chi^2 < \chi_\alpha^2(n)$, it is proofed that H_w is satisfied with $B(n,1/2)$, and correspondingly, plaintext and cipher text are independence, otherwise, they have relationship.

During experiments, we regard known cryptographic algorithm A(original SBOX in AES algorithm) as plaintext, and any generation behind A called A' as cipher text. The results of experiments are shown as table 2.

SBOX in A' (decimal representation)

167，204，144， 11，225，182， 57，159，209，231， 32，109， 30， 90，
37，215，
33，236，227， 40，214， 97， 98， 43，220，222，124， 71，198， 81，
174， 21，
210， 48， 67， 26，181， 17，106， 50，149，171，130，237，228，122，
148，128，
108， 28， 12，154， 85，189，186，176，129，249，136，247， 47，178，
217，234，
161，142，162， 42，197， 80，194， 86，211，218，180，179，133， 22，
113， 35，
239，243，121，201， 56， 41，103，125， 75，199，244， 29， 96， 52，
34，100，
147， 58，104，200，251，143， 82，250，169，123， 44， 70， 1， 84，
160，172，
5，138， 7，207， 9，131， 27， 0，246，140，168， 69，116，141，
3，233，
76， 15，152， 45， 38，156，137，229，202，230，252，253， 19， 16，
78，184，
126，254，240， 91，101，146，115，193， 39，221， 92，102，135， 72，
8，139，
120， 99，163，166， 63， 87，127，132，183，110，238，111，188， 62，
73，173，
46，107， 51，190，157， 49，224，155， 4，158，203，118，205，255，
36， 20，
223，114， 83， 61， 60， 74，191， 64，177，196， 66，219，145， 65，
165，248，
119，212，185，134，213， 53， 55， 25，170，105，216， 24， 94，175，
31， 13，
151，153， 95， 89， 23，226，187，117，112，206，195，242，192， 68，
150，245，
235，164， 59，208， 2， 93， 79， 88， 18， 54， 10， 14，232， 6，
241， 77

Table 2: experiments of H_w between different evolutionary SBOXes

w	54	55	56	57	58	59	60	61
H_w	0	2	0	0	1	2	2	1
w	62	63	64	65	66	67	68	69

H_w	1	2	2	1	0	1	0	1

As for other values of w, H_w are all zero. Because χ^2 is 20.1 which is smaller than χ^2 at α =0.05 and α =0.1, SF_{gi} (plaintext) and SF_{gi+1} (cipher text) are proofed to be independent.

Table 3 shows values of χ^2 among different 10 generations of evolutionary SBOXes.

Table 3: experiments of χ^2 between different evolutionary SBOXes

SBOXes	χ^2 value
s0—s1	20.1
s0—s2	14.3
s0—s3	29.1
s0—s4	22.8
s0—s5	50.5
s0—s6	46.2
s0—s7	20.3
s0—s8	69.8
s0—s9	19.4
s1—s2	27.8
s2—s3	21.6
s3—s4	13.7
s4—s5	20.5
s5—s6	15.6

S0 is the original SBOX in AES algorithm, and S1…S9 are generated with evolutionary cipher[29]. Appendix shows the values of these evolutionary SBOXes. $\chi^2_{0.05}(128)$= 154.3015159 $\chi^2_{0.1}(128)$= 147.8048

We have made expensive experiments about evolutionary cipher, all of these experiments showed different results of evolutionary cipher have good independence, and evolutionary cipher can

effectively prevent forward and behind attacks.

5. Experiments of Evolutionary Cipher Effectively against DPAs

Different with last experiments, these experiments are trying to test the effectiveness of evolutionary cipher resistance against DPA attacks.

In order to decrease noise disturbing from test equipments and circuits, we adapted cyclone2 series FPGAs of Altera as test chips, DE2-70FPGA development kits and primepower of synopsys as test environments.

We designed three different levels of experiments to test the resistance of evolutionary cipher against DPAs.

5.1 TEST1

5.1.1 Test1 Measure

1）test purposes: When attackers have known every detail of D function whenever it changed, it tests whether evolutionary cipher can resistance against DPA attacks effectively.

2）parameters and conditions:

- Known: plaintext P, key Ks, and Ks keep unchanged during DPA attacks;
- Attackers can put correct Ks into differential power computation model[kocher]. And if it showed peaks with correct Ks, resistance is not success;
- Making expensive experiments to calculate success ratio.

3）Test1 Steps:

Step1: Randomly selects plaintext P and key Ks, and Ks keep unchanged during DPA attacks, P changed randomly in every test;

Step2: Power sample point is set at the end of first round of block cipher such as AES algorithm;

Step3: Using EDA simulation and power analyzer to collect certain information such as cipher text C_i and power $T_i[j]$ on line;

Step4: To put correct Ks into D function and compute value of b. The reason why we can use correct

Ks is that test1 supposes attackers can get every detail of cryptographic algorithm which means that attackers can get details of D function. If correct Ks is collected by attacker, the value of b is correct. The real purpose of test1 is to find whether does power peaks in simulation tools. In order to decrease the complexity of test, we directly bring correct Ks into D to compute correct b.

Step5: To compute differential power based on the model proposed by kocher, and judge whether the condition of DPA is satisfied.

Step6: Making expensive experiments to calculate success ratio.

5.1.2 Analysis of Test1

The follow of experiments is shown as figure 2. And the results are listed in figure 3 and 4.

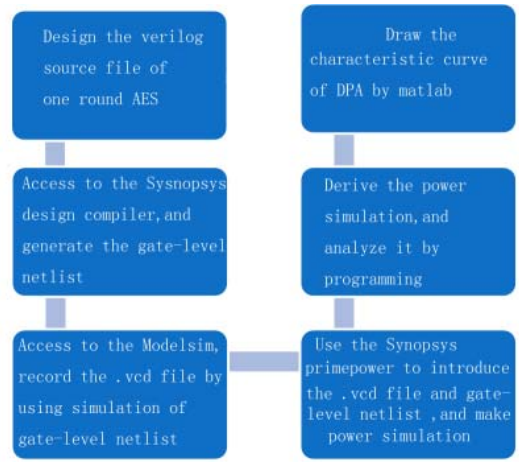


Figure2: the follow of experiments

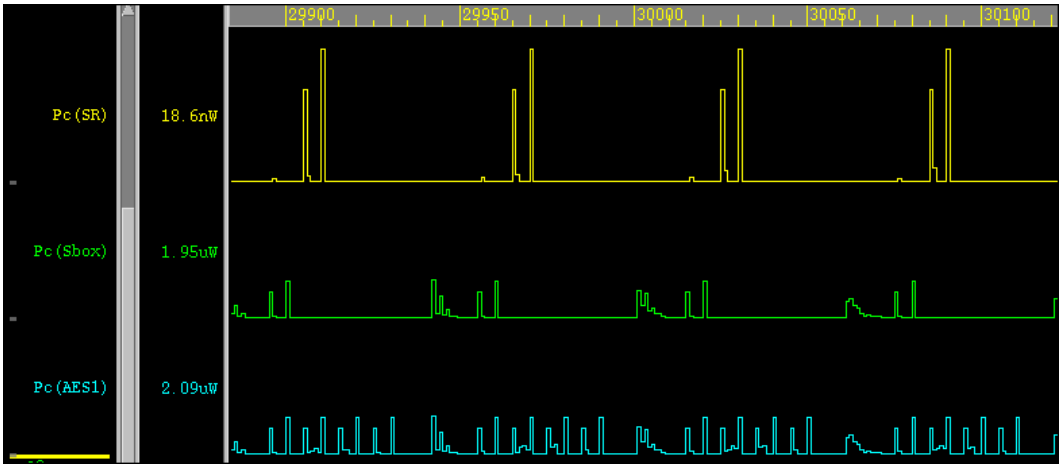


Figure 3: experiment results of test 1 in primepower

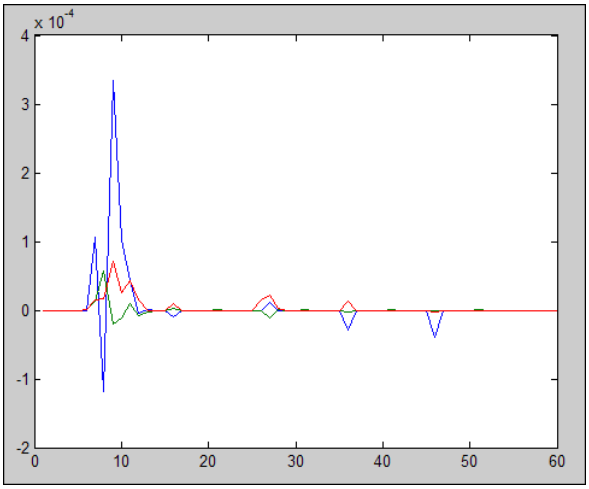


Figure 4: power cycles of test1 in matlab

Illustration:

Test	Number of samples	Time of test	Blue cycle (attackers know)	Green cycle	Red cycle	conclusion
Test1	10,000	60ns	Correct traces	Random traces	Originaltraces	DPA Success

The results of test 1 have shown that if attackers can get knowledge of every detail of D function or cryptographic algorithm, attackers can find blue cycle which represents correct traces, so DPA attacks can be success. Moreover, effectiveness of DPA dependents on the number of samples, more samples more effective. If the number of samples is less than 5000, correcttraces will be covered by random traces.

5.2 TEST2

5.2.1 Test2 Measure

1）test purposes: Attackers can find D function has been changed, but don't know the details of variety. Test2 evaluate the success ratio of evolutionary cipher resistance.

2）parameters and conditions: Attackers regards D as randomly changed and conditions are same as

test1.

3）Test2 Steps:

Step1: Randomly selects plaintext P and key Ks, and Ks keep unchanged during DPA attacks, P changed randomly in every test;

Step2: Power sample point is set at the end of first round of block cipher such as AES algorithm;

Step3: Using EDA simulation and power analyzer to collect only power $T_i[j]$ on line;

Step4: For attackers are not know the details of changed D, they regard D as random function, and then randomly select the value of b as each power cycle;

Step5: To compute differential power based on the model proposed by kocher, and judge whether the condition of DPA is satisfied.

Step6: Making expensive experiments to calculate success ratio.

5.2.2 Analysis of Test2

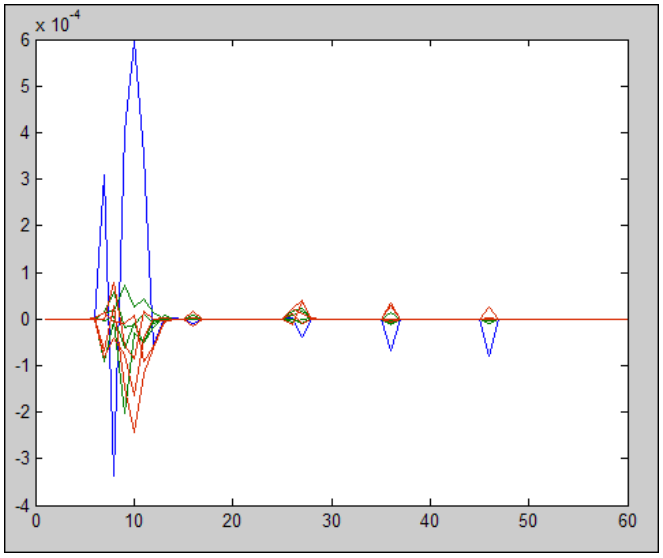


Figure 5: power cycles of test2 in matlab

Illustration:

Test	**Number of samples**	**Time of test**	**Blue cycle**	**Green cycle(attackers know)**	**Red cycle**	**conclusion**

Test2	**10,000**	**60ns**	**Correct traces**	**Randomtraces**	**Originaltraces**	**DPA Failure**

In test2, attackers can not get hold of correct traces, but get random traces, so the DPA attacks are fail.

5.3 TEST3

5.3.1 Test3 Measure

1）test purposes: Attackers don't find the variety of D function, and use original D function to compute differential power. Test3 tries to calculate the success ratio of evolutionary cipher resistance against DPAs.

2）parameters and conditions: Parameters and conditions are same as test1.

3）Test3 Steps：

Step1: Randomly selects plaintext P and key Ks, and Ks keep unchanged during DPA attacks, P changed randomly in every test;

Step2: Power sample point is set at the end of first round of block cipher such as AES algorithm;

Step3: Using EDA simulation and power analyzer to collect certain information such as cipher text C_i and power $T_i[j]$ on line;

Step4: For attackers have not found the variety of D function, test3 puts observed cipher text C_i and correct key Ks into D function to compute b corresponded with each power cycle.

Step5: To compute differential power based on the model proposed by kocher, and judge whether the condition of DPA is satisfied.

Step6: Making expensive experiments to calculate success ratio.

5.3.2 Analysis of Test3

The results of test3 are also shown in figure 5, and attackers can get red power cycle which represents original traces and has no relationship with correct traces, so the DPA attacks are fail too.

5.4 test 4

From previous tests, we can find that SBOX plays an important role in DPA attacks. If attackers can not get knowledge of SBOX, DPA attackers can not be success. However, for DPAs have timing property, if attackers can observe linear operations in one round of AES algorithm, they could not regard of SBOX. Test 4 are designed to test whether DPA attacks on linear operations can success. Results of test 4 are showed in figure 6 and 7.

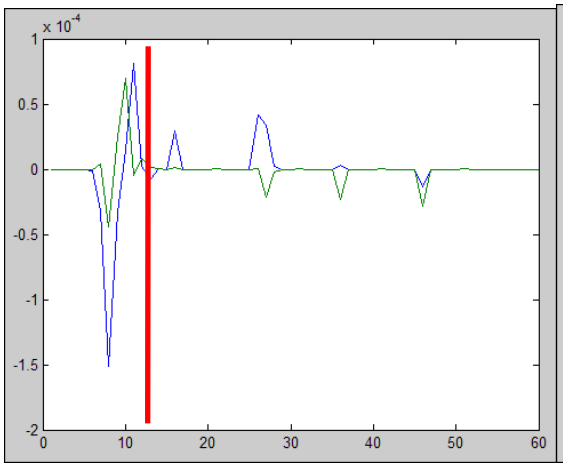


Figure 6: one round with SBOX

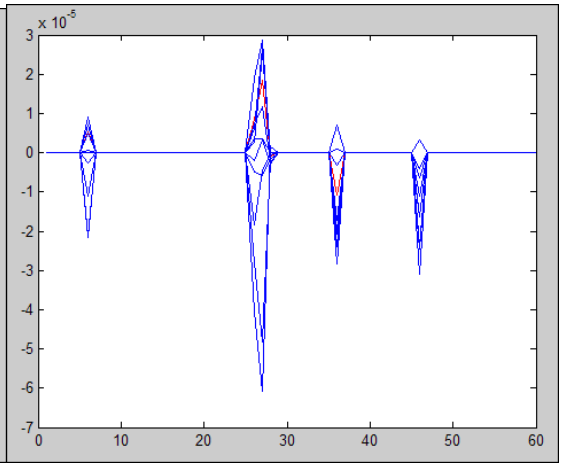


Figure 7:one round without SBOX

Figure 6 shows the results of one round implementation with SBOX, and the cycles behind red line represents the power consumed by linear operations. Blue and green cycles are correct and random traces respectively.

The maxmum value of blue cycles after red line smaller than the right peak about 10 times, so this case DPA attack is ineffective.

Figure 7 shows the results of one round without SBOX, and red and blue cycles are correct and random traces respectively. For correct cycles are all covered by random ones, DPA attack is failure, too.

So, DPA attacks can not be success on linear operations of AES algorithms.

5.5 Analysis of Tests

After security analyzes in session 4 and experiments in session 5, we can verify evolutionary cipher can effectively resist against DPA attacks from dynamic algorithm aspect. Comparing with some existing resistances, the purpose of evolutionary cipher is mainly focused on destroying differential power computation model in analyzing of DPAs. Using evolutionary cipher designing principles, cryptographic algorithms can effectively resist against DPAs through extensive experiments and are proofed to be a safety implementation to resist several attacks. Table 7 shows the combined results of different effective experiments.

Table 7: combined results of effective experiments

Test	**Number of samples**	**Time of test**	**Blue cycle**	**Green cycle**	**Red cycle**	**conclusion**
Test1	**10,000**	**60ns**	**Attacker know**			**DPA Success**
Test2	**10,000**	**60ns**		**Attacker know**		**DPA Failure**
Test3	**10,000**	**60ns**			**Attacker know**	**DPA Failure**

Bule, red, green cycles are represents correct, random, and original traces respectively.

The reason why evolutionary cipher can effectively resist against DPA attacks is its dynamic structure controlled by TRNG which can generate both safety and unpredictability of random sequence. Safety random sequence generated by TRNG meets the Gaussian distribution and makes different power almost same in division of one or zero. Moreover, unpredictability of TRNG prevent attackers to get the details of correct algorithm and then can not acquire correct division of one and zero. This difficulty was also introduced in the research of DFA[31].

However, it should be particularly pointed out that TRNG is the key point of evolutionary cipher but is not as long as TRNG controls variety of algorithm can be against DPAs or other SCAs. Moreover, through extensive experiments, we find that only using TRNG to control the variety of evolutionary cipher can not fit the demand of security of cryptographic algorithm. As we introduce in section 4 and 5, safe TRNG must generate long enough random sequence and evolutionary objects must have large enough size. Focused on the base of DPA analyzing, this paper utilizes dynamic structure of evolutionary cipher to resist against DPAs. From this point of view, we combined TRNG, intelligent searching algorithm and cryptographic designing principles to automatically design cryptographic algorithms and evolutionary cipher can not only resist DPA attacks but several mathematic analyzes.

6. Conclusion

This paper proposes a new countermeasure called evolutionary cipher against DPA attack which is one of the most threatening in SCA attacks. It is firstly to combine intelligent computation method and cryptographic designing principles in design dynamic structure algorithm in order to resist DPA attacks. The advantage of evolutionary cipher is it can also resist several mathematic analyzes besides DPAs. Evolutionary cipher provides a new idea for design and implementation of cryptographic algorithm which can achieve both cryptographic and physical security.

Evolutionary cipher is to destroy the differential power computation model proposed by kocher. Moreover, we have quantitatively analyzed effectiveness and security of evolutionary cipher against DPA attacks, and concluded designing principles which can also be learned by other dynamic structure cryptographic algorithm to resist DPA attacks.

We will continue to deeply study designing methods and theory of evolution cipher, and the results will be applied to resist more SCA attacks and mathematic analyzes. At the same time, we will make an intensive study of the dynamic aspects of the algorithm.

Appendix：

Appendix1:

Evolutionary SBOXs tested in part 5.

s0:

99，124，119，123，242，107，111，197，48，1，103，43，254，215，171，118，
202，130，201，125，250，89，71，240，173，212，162，175，156，164，114，192，
183，253，147，38，54，63，247，204，52，165，229，241，113，216，49，21，
4，199，35，195，24，150，5，154，7，18，128，226，235，39，178，117，
9，131，44，26，27，110，90，160，82，59，214，179，41，227，47，132，
83，209，0，237，32，252，177，91，106，203，190，57，74，76，88，207，
208，239，170，251，67，77，51，133，69，249，2，127，80，60，159，168，
81，163，64，143，146，157，56，245，188，182，218，33，16，255，243，210，
205，12，19，236，95，151，68，23，196，167，126，61，100，93，25，115，
96，129，79，220，34，42，144，136，70，238，184，20，222，94，11，219，
224，50，58，10，73，6，36，92，194，211，172，98，145，149，228，121，
231，200，55，109，141，213，78，169，108，86，244，234，101，122，174，8，
186，120，37，46，28，166，180，198，232，221，116，31，75，189，139，138，
112，62，181，102，72，3，246，14，97，53，87，185，134，193，29，158，
225，248，152，17，105，217，142，148，155，30，135，233，206，85，40，223，
140，161，137，13，191，230，66，104，65，153，45，15，176，84，187，22

s1:

167，204，144，11，225，182，57，159，209，231，32，109，30，90，37，215，
33，236，227，40，214，97，98，43，220，222，124，71，198，81，174，21，
210，48，67，26，181，17，106，50，149，171，130，237，228，122，148，128，
108，28，12，154，85，189，186，176，129，249，136，247，47，178，217，234，
161，142，162，42，197，80，194，86，211，218，180，179，133，22，113，35，
239，243，121，201，56，41，103，125，75，199，244，29，96，52，34，100，
147，58，104，200，251，143，82，250，169，123，44，70，1，84，160，172，
5，138，7，207，9，131，27，0，246，140，168，69，116，141，3，233，
76，15，152，45，38，156，137，229，202，230，252，253，19，16，78，184，
126，254，240，91，101，146，115，193，39，221，92，102，135，72，8，139，
120，99，163，166，63，87，127，132，183，110，238，111，188，62，73，173，
46，107，51，190，157，49，224，155，4，158，203，118，205，255，36，20，
223，114，83，61，60，74，191，64，177，196，66，219，145，65，165，248，
119，212，185，134，213，53，55，25，170，105，216，24，94，175，31，13，
151，153，95，89，23，226，187，117，112，206，195，242，192，68，150，245，
235，164，59，208，2，93，79，88，18，54，10，14，232，6，241，77

s2:

5，76，187，139，159，240，177，4，202，7，84，206，119，233，188，6，
204，15，13，130，254，11，155，8，37，55，109，0，21，19，29，26，
47，89，216，141，106，78，235，247，50，92，46，68，191，3，179，169，
156，209，137，140，48，58，246，117，225，165，128，232，154，234，248，2，
182，244，107，59，111，151，51，146，239，69，180，201，110，193，16，105，

189，243， 66，142， 63，194， 27，250， 72，207，211，251，172，113， 49， 83，
121，116，149，122， 98，186，129，168，162，227，229， 71，167， 22，123， 45，
 62， 88，131， 85， 86， 94，236，   1，126，152，148，173，133，150，170，161，
 38， 12， 99，242，220， 30， 42，238， 93，174，215，200，103， 95， 96，120，
134， 25，226， 40， 73，217， 41， 57，   9，104， 32，124，138，171，237，102，
127， 24，175，198，255， 54， 20， 74，241， 52，199， 28， 90，212， 80，245，
100，135，125， 53，147，184， 70， 97，114，213，221，101，192，118，223， 35，
203，249， 56，143，231，164，158，160，253， 64，224， 43，252，144，183，163，
 18，218，145， 81， 17， 67， 10，196， 79，197，166， 87，185，176，205，195，
 77， 65， 36，219， 14，190，214，210，153， 75， 31，178， 44， 61， 91，181，
 39， 60， 33， 23，157， 82，108，132，136，115，208， 34，228，222，230，112

s3:

164，113，250，180， 14，111，167，132，124，228， 86，252， 66，104， 26，196，
188，249，185，137，178，121，142， 25，208，170， 37，   4，158， 94，131， 99，
141，235，   6，116，197， 49， 40，143， 10， 75，173，108，122，100，231，128，
110， 59，244， 84， 74， 23，175，   2，117， 29，201， 72，174，   8，114， 68，
 71，239，229， 55，101， 19， 42，179，168， 76，   7， 28， 69，   1， 62，165，
 58， 15，172， 20，183， 97， 67， 50，241，220，123， 18， 32，130，106，182，
159， 34， 83，255，216，218，233，160，253， 53，245， 12， 93，254，223，205，
151，203，169，118， 22， 11，200， 36，127，238，115，   0，105， 51，224，157，
176，153，248， 47，134，227， 45，136，107， 96，251， 60，120， 43，152，191，
  9，   3， 21，109，209， 38， 77，119， 57，133，112， 63，148，192，232， 88，
 95， 35， 64，225，146，138，190，177， 79，202，193，163，139，155，214，207，
 24， 41， 31，234，147，154， 44，184，226，187，166， 56， 33， 98，230， 16，
 92， 82， 87， 52，181， 61， 46，189，210，236， 85， 13，242，243，103，221，
126， 70，211，246， 30，140， 89，161， 17，129，125， 54，186，135，156， 65，
 81，204，240，102，217， 90，219， 91，206，145，195，199，237，247，171， 39，
144，215， 80，222， 78，150，   5， 73，212，194， 27， 48，213，198，149，162

s4:

170，217，207，234， 56，161，166， 42，237，183， 69，215， 21，189，104， 55，
202，195，222， 30，242，249，   2，100，103，146，148， 16， 66，101， 54，145，
 14，139， 24，205， 51，196，160，   6， 40， 49，142，173，245，141，187， 58，
165，236，247， 77， 53， 92，134，   8，201，116，   3， 61，130， 32，213， 13，
  1，155，179，220，137， 76，168，246，154， 45， 28，112，   9，   4，248，174，
232， 60，138， 80，230，153， 17，200，227， 87，241， 72，128， 50，181，226，
 70，136， 81，219， 71， 79，131，186，211，212，243， 48，105，223， 91， 19，
102， 11，158，197， 88， 44，   7，144，225，159，209，   0，185，204，167， 78，
250， 94，199，188， 34，171，180， 26，177，157，203，240，253，172， 90，198，
 36， 12， 84，169， 99，152， 41，193，228， 46，221，252，106， 39，135，125，
 97，140， 29，163，114， 18，194，254， 33， 15， 35，182， 22， 86，127， 27，
 96，164，124，143，118， 82，176，218，175，214，162，224，132，149，191， 64，
109， 85， 65，208，238，244，184，206，111，151， 73， 52，239，235，129， 83，

229， 5，107，255，120， 10，121，190， 68， 62，233，216，210， 38， 74， 25，
89， 23，231，133， 67，117， 75，113， 31，126， 43， 59，147，251，150，156，
122，123， 93， 95， 37， 98， 20， 57，119， 47，108，192，115， 63，110，178

s5:

146， 67， 27，143，224，190，162，168，147，230， 9，123， 84，206，189，220，
15， 43， 95，120，239，195， 8，141，129，114，106， 64， 21，137，216，126，
56， 22， 96， 19，204， 55，186， 24，160，196， 2，142，243， 14，214，232，
174，151，251， 41，212，109， 34， 32， 3，205， 12，244， 50，128，115， 52，
4， 86，246， 87， 30， 45，154，255， 82，180，112，221， 36， 16，199，130，
135，240， 18， 93，191， 94， 68， 7，171， 65，227， 61， 58，200，238，175，
5， 26， 89， 75， 1， 33， 54，210，107，119，235，192，185， 91，113， 76，
133， 44， 66， 51，125，176， 28，122，163， 70， 99， 0，222， 23，166， 37，
207，101， 59，202，136，150，234，104，254， 78， 11，231，211，138，117， 63，
144， 48， 77，158，145， 90，164， 35，183，184， 83，215，181，156， 38，233，
153， 10，116，182，213， 72， 47，223，132， 60，140，226， 88， 69，225，108，
157，170，237， 6，197， 85，250， 79，134，127，178，167， 42，110，198， 29，
169， 73， 25，103，159，247，218， 31，161，102， 57，208，155，139， 62， 81，
179， 20，177，219，253， 40，249，194， 13，248，131， 71，111，152， 53，100，
121， 92，187， 46， 17，201， 49，217，124，229，172，236，118，203， 98， 74，
245，241，105， 97，148，149， 80，228，193，188，173， 39，209，252，165，242

s6:

46，145，224， 61， 59， 44， 10， 12， 38，170， 23，153，166，245， 16， 84，
186， 33，148，144，181，216，108， 70，236， 47， 75， 64， 56，184，102， 11，
156， 25，195， 41，127，173，130， 17，171，185，240， 88， 29， 69，128， 72，
217，158， 83，211，137，229， 22，155， 6， 21，225， 19，135，121，116，207，
152，221，203，104，210，125，179，160， 7，209，228， 36，244，189，150，197，
50，233，254， 91，141，241，175， 37，212，133， 98，107， 31，134，247， 40，
113， 43，139，253，208，164，180，114， 4，106，101，117， 60， 52， 15， 45，
151， 26，249，222， 82，192，132， 39， 54，103，214，162， 28，177，143， 76，
55， 80， 35，239，252，157，167，250，193，165，140，146， 58，119， 24，120，
66， 8，105，109， 51，219，169， 92， 0，201，110， 30，213， 94，174， 97，
67， 96，172，123，232，147，168， 74，138，218， 85， 48， 5，223， 14，198，
163，196， 86，111， 99，237， 18，122，230，129， 32，176， 73，188，200， 20，
136， 63， 53，142， 90， 79， 77，205， 27， 1， 95，149， 9， 93，100， 42，
71，231，251，131，246， 49，183，199， 89，238，194，242，215，235，118，204，
243， 57，227，206，115， 3，178，161， 78， 62， 87， 13，124，202，191，220，
159， 2，154， 65，248，112，226，255， 34，190， 68，187，234， 81，182，126

s7:

107，180， 59，238，174，215， 79，126，222，228，145，187， 92， 7，144，203，
73， 47，138，193，246，183，130，102，156， 70，148， 57， 78，244，134， 43，
159，209，216，219，188，205， 46， 63，160， 44， 32，140，253，168， 15， 95，

139， 68，223，234，106，117， 94，118，231，241，176，236， 81， 20， 60，157，
54， 30， 28，178，151，185，245， 76，207， 86，128， 35，251， 23，227，116，
114，198， 13，225， 25， 71，112，190，161，199， 85， 87，255，213，165， 45，
77， 90，124，147，177，220， 55，129， 53， 1，155，194，189， 84，169，152，
248， 93，210，250，158， 4，111，133，150， 42，243，153，218， 97，197，121，
132，184，119，191，146， 89， 10，221，226，143， 21，136，103，206， 65，224，
49，131，122，115，104， 9，142，163，171，149，127，123，214，154， 40， 2，
204，196，125，254，162， 18，120， 6， 5，240，192， 56， 69， 66，170， 29，
11， 58，113， 64， 26， 22，109，201， 99， 41， 82，211，105， 27，182，181，
96，229， 24，164，195， 75， 67， 98， 36，242， 38， 62，252，212， 14， 50，
235， 88，202， 37， 3，100，137， 8，179， 34，217，172，237， 72，166，175，
33，232，230， 31， 80，108，135，101， 48， 16，173，186，141， 83，239， 17，
19， 51，249，247，167， 52， 39， 91，200，233，110， 74， 12，208， 61， 0

s8:

41，176， 88， 90， 79，174， 72， 10， 6，122， 89，185，142， 91，131， 20，
54，126， 15，205， 52， 73，158，145，212，236，110， 57，213，217， 68，181，
30，133，239，100， 25，172， 16，252，156，129，180，231，128，248，152， 78，
250，247，207， 33， 26，144， 45，204， 55，111， 2，228， 94，195，138， 0，
48， 22，251，136，121， 77，139，115， 92，160，201，171，155，101，103，170，
186，242， 64，235，215， 56， 83， 99，187， 42， 23，150， 40，183，234，157，
206，203，216，148，229， 3，179， 13，125， 21，226， 5， 31，132，230， 7，
241， 93，140，193，112，153，249，123，134， 1，253，189， 65，227，177，199，
178， 86，240， 49， 32， 75，197，218， 50，167， 11， 39， 59，135，238，141，
220，211，147，104，209，245，161，151， 96，102，137， 61， 9， 12， 70， 38，
58，106， 35， 53， 19， 17，210，237，127，120， 29， 84， 4， 24， 34， 37，
232， 67，159，188，198，225，202，130，224，109， 97，244， 43， 81，107， 60，
95，114，219，146， 28，105，184，255， 47， 69，117，143， 85，191，119，118，
36， 27，175，173，223， 80，124，164， 44，182，168， 46，196，222，254，149，
208， 71，163，200，165， 66， 76，243， 8，162， 14， 98，221， 82， 51，192，
246，214， 62， 63，116，166，233，108， 87， 74，190，169，113，194，154， 18

s9:

160，198， 4，106， 68， 66，149，232，118，234， 34，242，141， 73， 91，128，
111， 59，254，112， 87，161，180，214，213， 89， 54，147， 22，110，179， 41，
101，139，205，182，196，190，210， 58， 3， 72，229， 94， 67，116， 52， 39，
76，208，107，244，157，223，119， 9，243， 99，113， 93，148， 7，177，150，
173，195，100， 6，250，165，228，114，216，171，145，108， 2，194，226，175，
83，206，219， 40， 17， 86， 55，236， 32，156，146， 97， 21，122，251， 70，
8， 84， 37， 20， 16， 12， 92，239，222，142，143， 77，224，129， 50， 69，
131，238，199，178，130，144，227，120，162， 80， 42，240， 71，197，245，174，
33， 28，255，103，211，140， 18， 49，252，136，127， 64，126，247，235，217，
90，166， 48，189，105，123，221， 57，246， 53，172， 47， 60， 14， 45，176，
38，202，104，117，241， 62，169， 44，193，230， 36，121，102，218， 79， 25，

13，201， 51，135， 15， 65，186， 0，233， 74，133，137，155，207， 30， 43，
138， 75， 1，151，163，187， 78， 98， 10， 23，152，204， 96， 56，209， 95，
153，249， 31，154，115，212， 46，191， 29，203，183，253， 61，109， 85， 11，
200， 26，248，164，134， 88，158， 5， 24，215，237，231，185， 81， 27，220，
167，181，225， 82，170，132，192， 35，184， 63， 19，188，168，125，124，159

Appendix2:

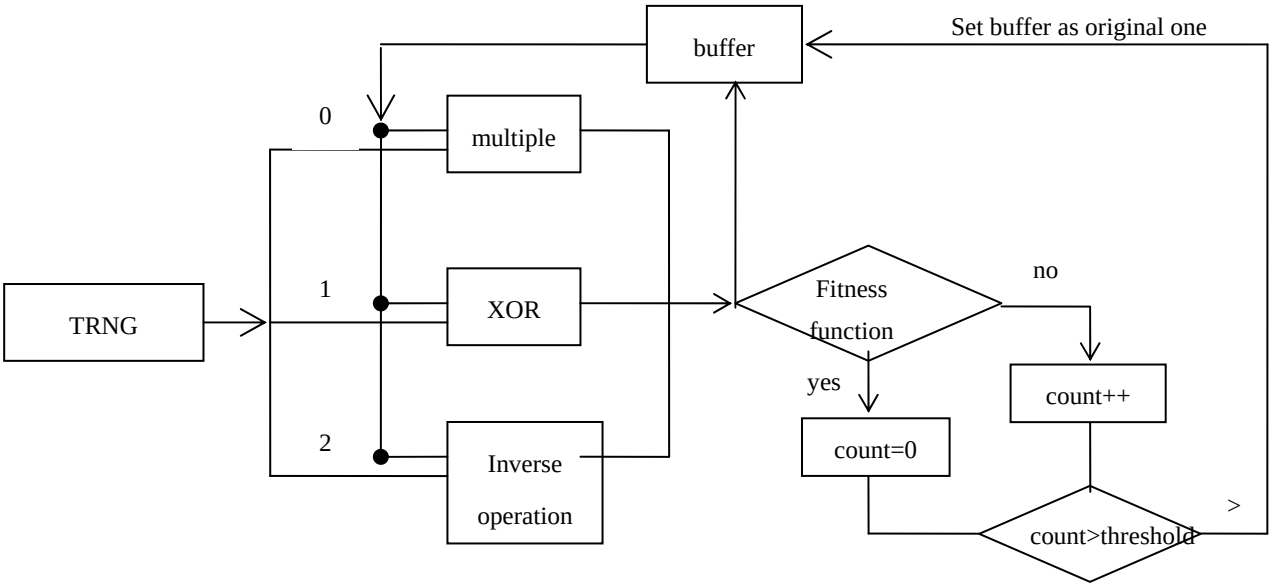
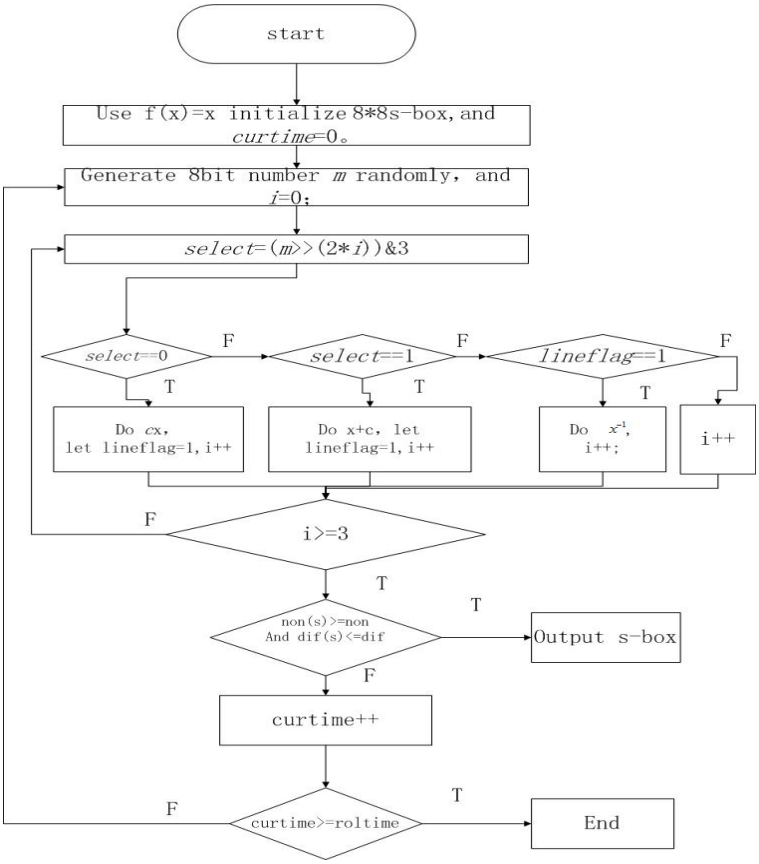


Figure 8: the structure of evolutionary algorithm[meng]



Fugure 9: The flowchart of the algorithm

Reference:

[1] A. J. Menezes, P. C. Oorschot, S. A. Vanstone. *Handbood of applied cryptography*(5th edition). CRC Press, 2001.

[2] Paul Kocher, Joshua Ja_e, and Benjamin Jun. Differential Power Analysis. Michael Wiener (Ed.): CRYPTO'99, LNCS 1666, pp. 388-397, 1999

[3] E. Biham and A. Shamir, Di_erential Cryptanalysis of the Data Encryption Stan-dard, Springer-Verlag, 1993.

[4] M. Matsui, \The First Experimental Cryptanalysis of the Data Encryption Stan-dard," Advances in Cryptology: Proceedings of CRYPTO '94, Springer-Verlag, August 1994, pp. 1-11.

[5] http://en.wikipedia.org/wiki/Side-channel_attack

[6] E. Trichina, D.D. Seta, L. Germani. Simplified Adaptive Multiplicative Masking for AES CHES 2002, LNCS 2523, pp.187-197, 2003.

[7] Tiri K，Hwang D，Hodjat A，et a1．AES—based cryptographic and biometric security coprocessor IC in 0．18

Ⅱm CMOS resistant to side channel power analysis attacks[J]．IEEE Journal of Solid—State Circuits(JSSC)，April，2006，41(4)：781—792．

[8] Nele Mentens, Benedikt Gierlichs, and Ingrid Verbauwhede, Power and Fault Analysis Resistance in Hardware through Dynamic Reconfiguration, E. Oswald and P. Rohatgi (Eds.): CHES 2008, LNCS 5154, pp. 346－362, 2008.

[9] Yang S，Wolf W，Vijaykrishnan N，et a1．Power attack resistant cryptosystem design： a dynamic voltage and frequency switching approach[c]． Proc．Of Design，Automation and Test in Europe (DATE 2005)，Munich，Germany，2005．64—69．

[10] Zhang Huan-Guo et al, Research on Evolutionary Cryptosystems and Evolutionary DES, CHINESE JOURNAL OF COMPUTERS 2003 Vol.26 No.12 P.1678-1684.

[11] Min Yang, Qingshu Meng, Huanguo Zhang, Evolutionary Design of Trace Form Bent Function. Available http://eprint.iacr.org. 2005/332.

[12] Qingshu Meng, Min Yang, Huanguo Zhang, Yuzhen Liu, Analysis of Affinely Equivalent Boolean Functions. The First Workshop on Boolean Functions and Application on Cryptography, De Rouen press, France, also available at http://eprint.iacr.org, 2005/025.

[13] Zhangyi Wang, Huanguo Zhang, Qingshu Meng, Differential Cryptanalysis of Hash Functions Based on Evolutionary Computing, ISICA 2007.

[14] Messerges T S，Dabbish E A，Sloan R H．Investigations of power analysis attacks on smartcards[C]．Usenix Workshop on Smartcard Technology, Chicago, Illinois, USA, M ay, 1 999, 151—162．

[15] S. Chari, C. Jutla, J. Rao, P. Rohatgi. Towards sound approaches to counteract power-analysis attacks. CRYPTO'99, LNCS 1666,pp398-412,1999.

[16] L. Goubin. A sound method for switching between boolean and arithmetic masking. CHES 2001,LNCS 2162,pp.3-15,2001.

[17] YongBin Zhou, DengGuo Feng. Side-Channel Attacks: Ten Years After Its Publication and the Impacts on Cryptographic Module Security Testing. This paper is invited to appear at the Physical Security Testing Workshop (Hawaii, September 26-29, 2005) held by the National Institute of Standards and Technology (NIST) of USA for discussions on issues specific to physical security testing and security requirements of cryptographic modules.

[18] Pramstaller N, Guerkaynak F K, Haene S, et a1. Towards an AES crypto—chip resistant to differential power analysis[c]． Proc． of the 30th European Solid—State Circuits Conference (ESSCIRC 2004)，Leuven，Belgium，2004.307—310

[19]Akkar M, Giraud C. An implementation of DES and AES.secure against some attacks[c]. Workshop on Cryptographic Hardware and Embedded Systems(Ches 2001). LNCS 21 62. Springer—Verlag, 2001. 309—31 8.

[20]Golic J D, Tymen C. Muhiplicative magking and power analysis of AES[C]. Workshop on Cryptographic Hardware and Embedded Systems (CHES 2002), LNCS 2523, Springer—Verlag. 2002.198—212．

[21] K. Tiri and I. Verbauwhede, A Logic Level Design Methodology for a Secure DPA Resistant ASIC or FPGA Implementation, In Proc. of Design Automation and Test in Europe Conference, pp. 246-251, 2004.

[22] E. Trichina, Combinational Logic Design for AES SubByte Transformation on Masked Data, Cryptology ePrint Archive, 2003/236, 2003.

[23] Tiri K，Verbauwhede I．A digital design flow for secure integrated circuits[J]．IEEE Transactions on Computer Aided Design ofIntegrated Circuits and Systems(TCAD)，2006，25(7)：1197—1208．

[24] Tiri K，Hwang D，Hodjat A，et a1．AES—based cryptographic

and biometric security coprocessor IC in 0．18Ⅱm CMOS resis—
tant to side—channel power analysis attacks[J]．IEEE Journal of
Solid—State Circuits(JSSC)，April，2006，41(4)：781—792．

[25] Daisuke Suzuki, Minoru Saeki, and Tetsuya Ichikawa. DPA Leakage Models CMOS Logic Circuits. CHES 2005, LNCS 3659, pp. 366–382, 2005

[26] Xilinx. OPB HWICAP,
http://www.xilinx.com/bvdocs/ipcenter/data sheet/opb hwicap.pdf

[27] www.altera.com/literature/hb/qts/qts_qii53013.pdf

[28] http://www.xilinx.com/bvdocs/ipcenter/data sheet/Xilinx Power Estimator User Guide.pdf

[29] State intellectural property office of P.R.C, application number: 200910060597.8

[30] C. E. Shannon, A Mathematical Theory of Communication*, Mobile Computing and Communications Review, Volume 5, Number I

[31] Amir Moradi,Mohammad T.Manzuri Shalmani,and Mahmoud Salmasizadeh.A Generalized Method of Differential Fault Attack Against AES Cryptosystem. CHES2006,LNCS4249,pp.91– 100,2006.

[32] http://techcenter.dicder.com/2006/0326/content_150.html