Research on Time Randomization based on soft interrupt Against DPA Attack

Lin Chen[1, a], Xingyuan Chen[1, b], Jinfu Xu [1,c], Moran Li [1,d]

[1]Institute of Information Science and Technology, Zhengzhou 450000 – China

[a]chenlin916@163.com, [b]chxy302@vip.sina.com, [c]xujf@mail.xd-soc.com, [d]465531933@qq.com

Keywords: elliptic curve cryptography; cryptography processor; time-randomizing; differential power analysis; software interruption.

Abstract. In this paper, a software interruption-based method is proposed so as to improve ECC processor's capability against DPA attack. Firstly, we analyze the principium of DPA-resilient time-randomizing. Secondly, combined with operational characteristics of ECC processor, a time-randomizing circuit is put forward. Finally, a power consumption simulation platform is raised for the purpose of making simulation analysis of the proposed circuit. The result shows that our product is secure against DPA attack and is hard to be excluded by differentiation functions aiming at power consumption track simultaneously, which meets the demand of DPA-resilient ECC processor.

Introduction

The security of elliptic curve cryptography [1] (ECC) is based on the hardness assumption that the discrete logarithm problem on elliptic curves is unsolvable by PPT algorithms, which makes ECC the most secure public key[2] encryption scheme with the same key length so far [3]. However, not only algorithms and protocols, but also the implementation plays an important role when implementing on hardware. Cryptographic devices may have power consumption disparities when processing different data. Differential power analysis [4] (DPA) attack makes use of this characteristic to analyze the devices' power consumption of specific moment statistically, which allows attackers to get secret information when performing operations without solving mathematical problems. Thus it is the most threatening among all the power attack methods.

By inserting randomized time delay while running the device, time-randomizing harasses the relation between power consumption, operation, and data, makes statistical analysis failed to resist DPA attack. The most widely-used time-randomizing technology at present includes multi-clock [5], gated clock delay [6], and redundancy command delay. Multi-clock requires multiple clock sources in the system, thus increase the control complexity of the system. The performance of random delay generated by gated clock delay on the energy trace is a straight line, which is easily excluded by differentiation functions[7]. The multi-cycle operation instructions with different period used in the system results in large basic granularity of time delay, and is therefore hard to be controlled.

For the inadequate listed above, based on studies the principles of anti-DPA technologies and binding with operational characteristics of ECC processor, we design a software interruption-based time-randomizing circuit that can resist DPA attack, and build a power consumption simulation platform to verify its feasibility.

Analysis of principium why time-randomizing is DPA-resilient

Let T[i][j] be the power consumption sampling signal while the cryptographic device is working properly, where i be the sampling sample, and j be the sampling point. We can know from the working principle of DPA attack that it needs N samples with N large enough. Define a set of differentiation functions that divided sampled data into two groups, T0={T[i][j]|D[i][j]=0} and T1={T[i][j]|D[i][j]=1}. T0 and T1 contains approximate elements when N is sufficiently large, which is defined as m. Then the differentiation consumption in point j can be denoted as

$$T_D[j] = \frac{1}{m}\sum_{i=1}^{m} T[i][j]|_{T[i][j]\in T_0} - \frac{1}{m}\sum_{i=1}^{m} T[i][j]|_{T[i][j]\in T_1}$$

And its expectation is expressed as

$$E(T_D[j]) = E(T[i][j] \mid D[i][j] = 0) - E(T[i][j] \mid D[i][j] = 1)$$

Suppose the random delay △ t has w possible values, with possibilities pk if the value equals k. Thus the sampling signal after inserting delay can be written as

$$T^*[i][j] = T[i][j-k]$$

Its expectation is equivalent to

$$E(T_D^*[j]) = \sum_{k=0}^{w} p_k E(T_D[i][j-k])$$

If the attacker guess the key right, the differentiation consumption curve will have reached its peak (let the value be A) at time n. Since other points have no relation with differentiation function, their differentiation consumption is about 0, thus we have

$$E(T_D[j]) = \begin{cases} A, j = n \\ 0, j \neq n \end{cases}$$

If the random delay is added, the peak will scatter to w+1 places, with pkA being the consumption of corresponding point, therefore

$$E(T_D^*[j]) = \begin{cases} p_k A, j = n+k \\ 0, j \neq n+k \end{cases}$$

Taking uniform distribution as an example, namely 1/(w+1). In this case, there will be (w+1) peaks with A/(w+1) each. If this value is less than the mean square error of noise, peaks will be covered, and DPA attack will fail. As the mean square error differs along with the application environment, we need to choose a proper range for w according to actual conditions.

Design of time-randomized ECC processor

Drawing on the idea of soft interruption [8], we introduce randomized time delay by executing interruption instructions during the operation.One important factor of the design of time-randomizing is to make the random delay controllable, which is related to both the resistance efficiency and impact on computing performance.

The ECC processor has two major features. The first is that there are few main operation modules as the calculations are mainly based on arithmetic over finite field, including modular multiplication, modular inversion, modular addition, and modular subtraction. Long period of computing instruction is the other trait of the ECC processor, and it varies with data length. The table below shows numbers of clock cycles of basic arithmetic units with different curve length.

Table 1. Cycles of basic operations with different curve length

Length	192bit	224bit	256bit	384bit	521bit
Modular multiplication	81	96	111	171	246
Modular inversion	920	1187	1355	2027	4177
Modular addition	4	5	5	5	6

These characteristics bring difficulty to the time-randomized control, and controls using the number of instructions as the basic unit is unsuitable accordingly. On the one hand, long period operations will have great impact on the performance of processors. On the other hand, the period differs by operation and curve length, therefore we can't manage the moment to insert the delay precisely.

Taking the fact that the randomized redundant instruction has no practical significance on cryptographic operations, we devise the following random delay control unit, using the clock cycle as the basic particle size, which is shown in figure 1. When executing the interruption program, it generates a random number as the counting period of delay control counter. When the counter ends, it generates a carry signal to control the return of the interruption: first, it is involved in the generation of the ending signal of the arithmetic module "Valid" to stop the redundant operation being executed. Second, it is used to generate the enable signal "PP_EN" for the stack register as well as the address selection signal "SEL" for the program counter "PC" to let the system return to the breakpoint and perform normal procedure. The value of the random number is equals to the period that the interruption is executed, namely the delay time of time-randomization.

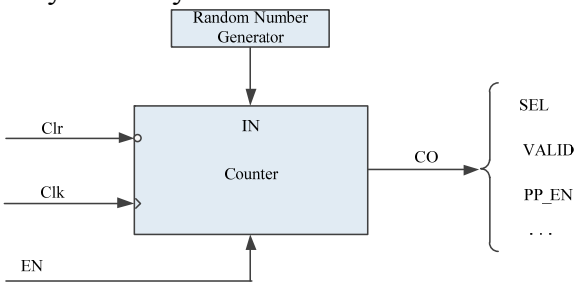

Fig 1. The logic of the control of randomized time delay

The time-randomization designed above can ensure the consistency of power consumption features between interruption parts and normal operations, making it indistinguishable in a single energy track. Unfortunately, DPA attacks make analysis on numerous energy track, which can resolve the randomized parts if their first addresses are the same, as is stated in figure 2.

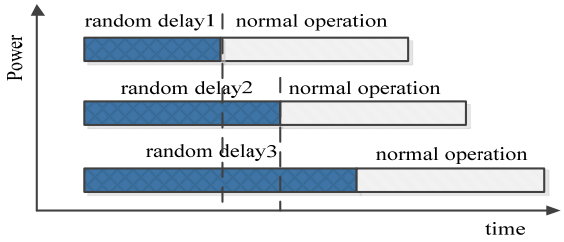

Fig 2. Schematic diagram of comparative analysis on energy tracks

To solve this problem, we add a controllable random delay address to the first address of the redundancy instruction to get the interruption address, as is shown in figure 3. The interruption part obtained this way is difficult to be excluded by differentiation.

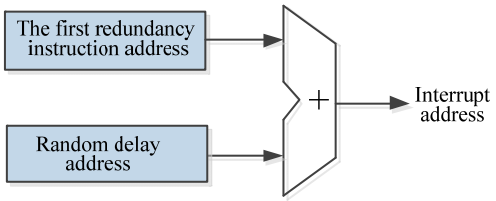

Fig 3. Schematic diagram of random interruption address

The randomized time delay resolves how to control the interruption time flexibly, while the randomized interruption address settles the consistency of power consumption between randomization and normal operations. In this paper, we propose a stage-variable pseudorandom generator, and configure its stage according to practical needs to control the range of random delay and offset address. On this basis, we bring out ECC processor control structure with software interruption, and design the software interruption command to support time-randomization against DPA attacks, which is revealed in Figure 4.

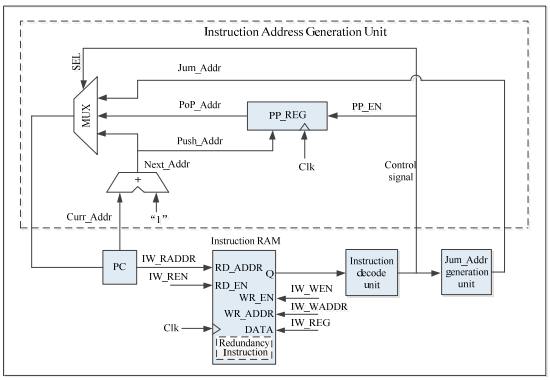

Fig 4. ECC processor control structure with software interruption

Verification of DPA-resilient capability of time-randomizing

The design of DPA-resilient time-randomizing needs random delay in each operation, and the interruption part should not be distinguishable in a single energy track. Thus the verification is main on the following three aspects:

1)Use simulations and check the executing time of arithmetic operations to verify if the random delay occurs.

2)Acquire the power consumption curve while the processor executes cryptographic computation and observe if the curve is consistent.

3)Simulate DPA attacks and verify its DPA-resilient capability.

Simulation verification of random delay

Taking the Montgomery point multiplication algorithm[9] as an example, we start up the interrupt program during the execution of the algorithm. The results of any two simulations with same input are shown in figure 5.

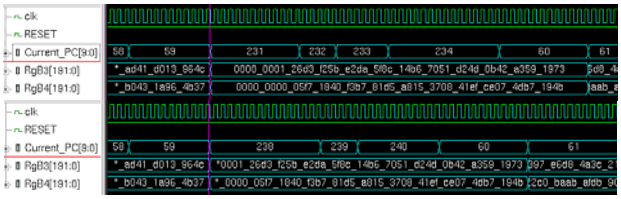

Fig 5. Results of software interruption-based time-randomization simulations

It can be revealed from the figure above that the value of "PC" ("current PC") turns to a different redundancy program address each time, and the return from interrupt time also changes, which meets the demands of jump address randomization and time-randomization.

1)Verification of consistency of single energy consumption track

The power consumption of the acquisition processor[10] when the software interruption execution algorithm is added is denoted in figure 6.

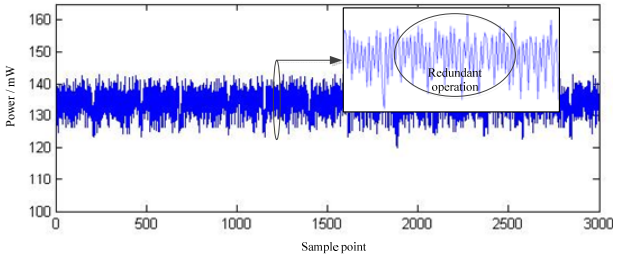

Fig 6. Simulation consumption of software interruption-based time-randomization

We can find that the power feature of the redundant operation part is consistent with normal operations, thus hard to be excluded by differentiation functions, fulfil the requirement of indistinguishability.

2)Simulation verification of DPA attack

In order to validate the capability of software interruption-based time-randomization against DPA attacks, we make a case study on the Montgomery point multiplication algorithm. When executing the software interruption program, the range of randomized delay are disposed to be 0 to 32 clock cycles and 0 to 64 clock cycles respectively, and we make a comparison to the system running normally, and the result can be found in figure 7.

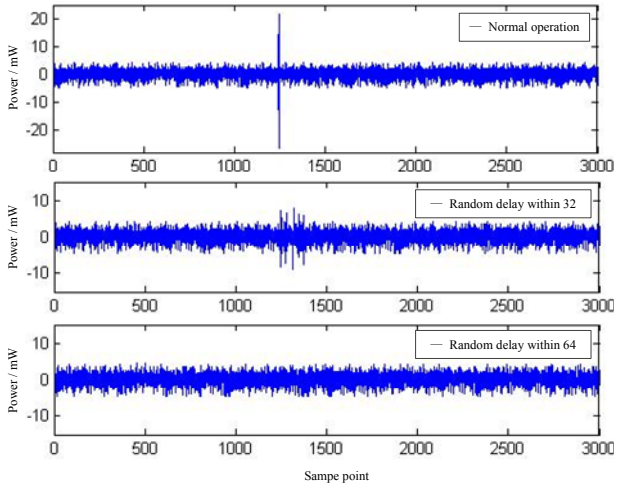

Fig 7. Results of DPA attacks with different time-randomization setting

It is apparent from the figure above that there is a significant peak in the first picture, where the system only does normal operations. If the delay is within 32 clock cycles, the summit drops evidently, but there exists an inevitable discrepancy from its theoretical value, which is supposed to be 1/32 of the former peak. The reason is that the operations are mostly in multi-cycle, the same data will be reserved for long time, and other influences including noises. When the delay is within 64 clock cycles, there is no obvious vertex on the power consumption curve, which is acceptable by the request of DPA-resilience.

Summary

In this paper, the principle that time-randomizing is effective against DPA attack is being analyzed, and binding with the processing features of ECC processors, a software interruption-based time-randomizing circuit is raised. Moreover, a power consumption simulation platform is proposed, and the capability of the circuit to withstand DPA attacks is being validated. Our design is on the level of processors, being regardless of specific algorithms, which makes it a DPA-resilient methodology with strong versatility.

References

[1] Victor Miller. Use of elliptic curves in cryptography [A]. In: H. C. Williams. Advances in Cryptography-CRYPTO'85[C]. Heidelberg: Springer-Verlag, 1986:417-426.

[2] He Liu, SPA Attack on ECC Implemented on MCU[J]. Journal of Chengdu University of Information Technology,2011,26(1):1-4.

[3] Le Ni. Research of Modular Multiplier Based on Normal Basis in Elliptic Curve Cryptography[D]. Zheng Zhou: PLA Information Engineering University,2013.

[4] Stefan Mangard, Elisabeth Oswald, Thomas Popp. Energy Analysis Attack[M].Dengguo Feng, Yongbin Zhou, Jiye Liu. Beijing: Science Press, 2010.

[5] DUAN Er-peng,YAN Ying-jian,LI Pei-zhi. Correlation electromagnetic analysis against AES cryptographic on implementations of FPGA[J]. Computer Engineering and Design. 2012, 33(8): 2926-2930.
[6] Bucek J, Novotny M. Differential Power Analysis under Constrained Budget: Low Cost Education of Hackers[C]//Digital System Design (DSD), 2013 Euromicro Conference on. IEEE, 2013: 645-648.
[7] AKIHIKO SASAKI and KOKI ABE. Algorithm-level Evaluation of DPA Resistance to Cryptosystems[J]. Electrical in Japan, Vol. 165, No. 3, 2008, pp. 1221-1228.
[8] Zhongmei Ma, Guangyun Ma, Yinghui Xu, Ze Tian. Base to ARM Embedded Processor Architecture and Application [M]. Beijing: Beijing University of Aeronautics and Astronautics Press, 2002.
[9] Darrel Hankerson, Alfred Menezes, Scott Vanstone. Guide to Elliptic Curve Cryptography[M]. Huanguo Zhang,trans. Beijing: Publishing House of Electronics Industry, 2005.
[10] Ito H, Shiozaki M, Hoang A T, et al. efficient DPA-resistance verification method with smaller number of power traces on AES cryptographic circuit[C]//Digital System Design (DSD), 2012 15th Euromicro Conference on. IEEE, 2012: 735-738.