Encryption Algorithm of Automatic Single Chip Computer and Embedded System Based on Big Data

Jiawei Chen1,*

1North University of China, Shuozhou, Shuozhou, 030051, Shanxi, China

*Corresponding author e-mail:1721084119.@st.nuc.edu.cn

Abstract. With the rapid development of network technology and communication technology, modern society is rapidly transforming towards an information society based on big data. The single-chip microcomputer and embedded system have received widespread attention from all walks of life since they came out. Now it has fully penetrated into people's daily work and life. In this regard, it is of great significance to study the encryption algorithms of automated single-chip microcomputers and embedded systems based on big data. The purpose of this article is to study the encryption algorithm of the automatic single-chip microcomputer and embedded system based on big data. This article adopts research methods such as literature data method, mathematical statistics method, and logical analysis method to study the encryption algorithm of automatic single-chip computer and embedded system based on big data. It mainly optimizes and adjusts the existing data encryption algorithm and strengthens the information. Through the normal test and abnormal test of the data encryption system, it can be seen that the system can encrypt and store data, which can be said to realize transparent encryption and decryption in the process of reading and writing. Studies have shown that although the system's built-in function takes 5-6 seconds to open a file, the time it takes for the function to open a file is between 45-48 seconds, which takes a little longer. When reading the specified data from the specified file, the correct data can be read only by providing a consistent key. It can be seen that the system is equivalent to providing data security measures in two aspects: identity authentication and data encryption. This design scheme can effectively protect the data security of the system.

Keywords: Network Technology, Communication Technology, Single-Chip Microcomputer And Embedded System, Big Data

1. Introduction
Modern society is an information society, and information has become an important strategic resource in social development, and social informatization is also becoming an inevitable trend of global development [1-2]. The rapid development of information technology has triggered unprecedented changes in society, and it has also completely subverted people's traditional lifestyle [3-4]. In modern society, the position occupied by information security is becoming more and more important. Since the birth of the single chip microcomputer, relying on its own significant advantages such as low cost, strong ability to adapt to the environment, high reliability, flexible structure, and easy production, it has attracted widespread attention [5-6]. But the storage capacity of the one-chip computer is not big, and the restriction of this aspect of the PC is smaller. Therefore, the research on encryption algorithms for single-chip computers and embedded systems has great practical significance [7-8].

In the research on the encryption algorithm of automatic single-chip microcomputer and embedded system based on big data, many scholars have conducted research on it and achieved good results. For example, Nti RB, Ryoo K. A has launched a very powerful 16 One-bit microcontroller series-MCS-96 series [9]. This was followed by the introduction of the MCS-196 series, which greatly improved the lack of MCS-96 series (internal structure). After the 1990s, driven by huge demands for distributed control, flexible production, digital communications, and information appliances, the operating system developed rapidly. Venkata SH, Likita R D, Shameem S. lead the development of embedded microcontrollers (MCU) and embedded DSP processors with ultra-high integration, high precision, high speed, and low power consumption [10].

This article adopts the literature research method to search the literature and works related to "big data", "automatic single chip computer and embedded system", etc., to study and use the new technology for reference, and extract useful information from it; mathematical statistics: to experiment The results obtained are graphed and then analyzed to obtain effective results.

2. Encryption Algorithm of Automated Single-Chip Microcomputer and Embedded System Based on Big Data

2.1. ECC Algorithm

The elliptic curve is a curve equation. The general form of the equation is as follows:

$$y^2 + axy + by = x^3 + cx^2 + dx + e$$

(1)

In the ECC algorithm, the elliptic curve equation usually adopts the following form:

$$y^2 = x^3 + ax + b (a, b \in GF(P), 4a^3 + 27b \neq 0)$$

(2)

When a, b, and p of the curve equation are determined, it is not suitable for encryption.

This is because the graph of the curve is continuous, and the integer points in the first quadrant are useful, so it needs to be discretized. The discretized point set plus the infinity point O is recorded as $E_p(a, b)$. $E_p(a, b)$ is generated as follows: For each integer in $[0, p)$, calculate the equation: $x^3 + ax + b (mod p)$. Determine whether the result of the above equation has a square root modulo p. If there is, find
the square root; if not, it means that there is no corresponding point of the x on the elliptic curve.

Ep(a, b) graphically represents a group of discrete points, and the above addition is defined as follows: Let P \( (x_1, y_1) \), Q \( (x_2, y_2) \), R(x_3, y_3) and P \( \neq \)-Q

If P, Q \( \in \) Ep(a, b), P+Q=R, then

\[
X_3 \equiv \lambda^2 - X_1 - X_2 (modp) \\
(3) \\
y_3 \equiv \lambda(X_1 - X_3) - y_1 (modp) \\
(4)
\]

Among them:

\[
\lambda = \frac{y_2 - y_1}{x_2 - x_1} \quad (P \neq Q) \quad \text{or} \quad \lambda = \frac{3x_1^2 + a}{2y_1} \quad (P = Q) \\
(5)
\]

When encrypting the plaintext, first convert the plaintext into points on the graph. Assuming that the plaintext message is m (m \( \leq \) M), k is an integer (the value is generally 30-50), take k=30 as an example, to calculate a series of x:

\[
x = \{mk+j, j=0, 1, 2,...\} = \{30m, 30m+1, 30m+2,...\}. \text{ Until } x^3 + ax + b \text{ (modp)} \text{ is the square root, the point on the curve is obtained } (x, \sqrt{x^3 + ax + b}).
\]

Now use it to realize the El Gamal cryptosystem:

Select an elliptic curve, and generate Ep(a, b), embed the plaintext message m on the curve, and find the point P_m.

Select the generator G of Ep(a, b), and disclose Ep(a, b) and G.

User A selects n_A as the private key and P_A=\( n_A G \) as the public key.

If user B wants to send a message P_m to A, he can randomly select a positive integer k, and transform P_m as follows to generate ciphertext C_m, namely C_m=\{kG, P_m+kP_A\}.

When user A decrypts, he only needs to subtract the multiple of his private key and the first point from the second point in the ciphertext point pair to recover the plaintext. The transformation process is as follows:

\[
P_m + kP_A - n_A kG = P_m + k(n_A G) - n_A kG = P_m \\
(6)
\]

2.2. Algorithm Improvement
(1) Optimization plan one

The RotByte transform and SubByte transform in the key expansion algorithm are components that are easy to implement, and these components are still retained in the first improvement scheme. Through big data analysis, it can be seen that the reason for the strong correlation between consecutive round keys in the original extended algorithm is mainly because only when $i/N_k$ is an integer, $W_{i-1}$ undergoes cyclic shift, S-box replacement, A series of transformations of the XOR wheel constant, and the remaining 3 characters are directly XORed by the other two characters. Therefore, in the first improvement scheme, each word in the round key is subjected to a complex transformation, and then the result is XORed with itself and the round constant to obtain the word of the next round key. This improved method reduces the correlation between successive keystrokes. Even if a certain round of keys is obstructed, it takes $2^{128}$ times to generate the previous round of keys.

(2) Optimization plan two

In the first optimization scheme, the security of the key has been improved, but after theoretical deduction, the running time of the algorithm has increased compared with before. This is because the generation of the key for each wheel in the original algorithm only needs to go through a complex transformation, but it needs to go through 4 times after the improvement. The improvement of scheme one was not ideal, so the original algorithm was improved again, and the optimization scheme two was proposed.

In the original key expansion algorithm, $W_{i+1}$, $W_{i+2}$, $W_{i+3}$ are all generated in the same way, and this is where it is vulnerable. Now keep the other processes in the original algorithm unchanged, only change the generation method of $W_{i+1}$, $W_{i+2}$, $W_{i+3}$ is obtained by XOR of $W_{i-3}$ and $W_{i-2}$, and $W_i$ is obtained by XOR of $W_i + 1$ and $W_i + 2$, And finally $W_i + 3$ is obtained by XORing $W_i + 1$ and $W_i + 2$.

Compared with the original algorithm, the correlation between consecutive round keys of the improved algorithm is reduced. Even if an illegal intruder intercepts a certain round of keys, it is not very helpful to reverse the previous round of keys. Compared with the optimization scheme 1, the scheme two only undergoes one complex transformation, which overcomes the shortcoming of the scheme one that has too many complex transformations (4 times).

However, the second scheme still has shortcomings. Although the correlation between consecutive round keys is reduced, the correlation within the round keys is improved. According to the key expansion process of Scheme 2, as long as the first two words of a certain round of key are intercepted, the latter two words can be directly calculated. Therefore, for this shortcoming, on the basis of the second plan, the second plan was improved, and the third optimized plan was proposed.

(3) Optimization plan three

In the key expansion process of Scheme 2, $W_{i+2}$ is obtained by the exclusive OR of $W_i$ and $W_{i+1}$, and $W_{i+3}$ is obtained by $W_{i+2}$ and $W_{i+1}$. Or get, where the generation of $W_i$ and $W_{i+1}$ is related to the previous round of keys. From this, it can be concluded that in order to
reduce the correlation between the words in the round key, the key parameter $W_{i+2}$ gets started. Therefore, the optimization scheme 3 adopts an outer round cyclic shift method, that is, after generating the extended key according to scheme 2, from the second round of the key to the last round, the third word of each round of the key is shifted to the left Go to the third word of the previous round key (the second round key moves to the last round). In this way, even if the first two characters of a certain round of keys are intercepted, the latter two characters cannot be directly calculated.

In summary, compared with the standard algorithm, the improvement of Scheme 3 not only improves the security of the key, but also reduces the time spent on key expansion. Therefore, the improvement of Scheme 3 is ideal.

2.3. Improvements in Algorithm Structure

(1) Improvements in the encryption process

In order to optimize the structure of the encryption algorithm, the round function of the first 9 rounds is now improved, and the two steps of row shifting and column mixing are combined. Now suppose that the State after byte substitution is $S_1$, and the State before the key addition is $S_2$. After the calculation process of row shifting and column mixing, the combined formula looks more complicated, but it is actually careful. Observation shows that there are only 4 definite coefficients for the factor to be multiplied, namely 00, 01, 02, and 03. Modular multiplication with 02 can be obtained by shifting 1 bit to the left; modular multiplication with 03 can be reduced to two steps: first perform modular multiplication with 02, and then perform modular addition with itself. Therefore, the combined operation is still easy to implement. By adjusting the round functions of the first 9 rounds in the AES algorithm, the original 4 calculation components can be simplified to 3, and the structure of the algorithm is optimized.

(2) Improvements in the decryption process

1) Exchange InvShiftRow and InvByteSub transform

In reverse byte substitution, each byte in the state array is operated, and the content of the byte is replaced according to the inverse S-box look-up table, and the position of the byte in the array is not changed. The reverse shift operation only changes the relative position of the byte stored in the state array, and has no effect on the content of the byte itself.

2) Exchange AddRoundKey and InvMixColumn operations

Inverse column mixing and round key addition operations only change the content of the bytes stored in the state, and do not affect the order of the bytes. In the inverse column hybrid transformation, the transformation operates on each column of the state. If the key is also regarded as a sequence of words, then the objects of the AK and IMC operations are the same.
3. Experimental Research on the Encryption Algorithm of Automatic Single-Chip Microcomputer and Embedded System Based in Big Data

3.1. Research Objects

Computer systems and encrypted systems.

3.2. Research Content

The performance of the system will be measured through experiments, and the performance of the enopen() and enwrite() functions provided by the system will be tested. In order to better measure the performance of the system, test the open and read/write interfaces of the system under the same conditions, record the time it takes to open, read and write the same file, and compare the time used by the two interfaces. It also analyzes the reasons for the difference between the two interfaces when reading and writing, and comprehensively evaluates the performance of the system.

4. Experimental Research and Analysis of Encryption Algorithms in Automated Single-Chip Microcomputers and Embedded Systems Based on Big Data

4.1. Open File Time Measurement

Use the above optimization scheme three, the system open() function and the enopen() function provided by the encryption system to open the file. Consider 10 file opening operations as one test, and record the time used for each test separately and then take the average value. Use the open() function and enopen() function to perform the tests, and record the average time used to open the file in each test as shown in Table 1.

| Operating | First time(ms) | Second time(ms) | Third time(ms) | Fourth time(ms) | Fifth time(ms) |
|-----------|----------------|-----------------|----------------|----------------|---------------|
| Open()    | 5.9            | 6.1             | 5.5            | 5.4            | 5.6           |
| Enopen()  | 47.3           | 47              | 45             | 48             | 44            |
Figure1. Open file time comparison table

As can be seen from the data in Figure 1, the execution time for the system's built-in function to open the file is 5-6 seconds, and the time for the function to open the file is between 45-48 seconds. The difference in the time it takes to open a file between the two interfaces is about 40ms. Analyze the difference in file opening speed between the two interfaces, mainly because the key management module needs to be called to obtain the key in the process of using the enopen() function to open the file, and the file stream object needs to be dynamically allocated, which will cause some time delay.

4.2. Measurement of File Writing Time

When calling the enwrite() function to write a file, data encryption is required. Data encryption is a time-consuming operation. Test the performance of the enwrite() function when using two encryption algorithms. In order to better measure the performance of the encryption system, the same test is also performed on the write() function, and the test results are compared and analyzed.

Take the data volume of 1K, 100K, and 1M as examples to test the time used to execute the enwrite() function and the write() function using the two encryption algorithms. In the test process, every 10 write operations are regarded as a test, and the average time of each write operation is recorded. The results are shown in Table 2.

Table 2. File writing time comparison table

| Data Size | Execute write() | Execute enwrite() using DES encryption algorithm | Use the RC4 encryption algorithm to execute enwrite() |
|-----------|----------------|-------------------------------------------------|--------------------------------------------------|
| 1K        | 45             | 88560.7                                         | 9849.5                                           |
| 100K      | 2346.5         | 975677.8                                        | 114356.8                                         |
It can be seen from the data in Figure 2 that the speed of executing the `enwrite()` function is much slower than the speed of executing the `write()` function. This is mainly because encryption and decryption is a relatively time-consuming operation.

It can be seen from the above test data that the encryption and decryption operation is a relatively time-consuming operation. Taking into account the fact that the amount of general data in embedded systems is relatively small and there are certain requirements for data security, the system can effectively improve the data security of the system to a certain extent.

5. Conclusion

With the rapid development of microcomputers, single-chip microcomputers and embedded systems have penetrated into people's daily lives. However, the security of microcomputer information has caused people's concerns. In the process of transmitting information, they are often attacked by illegal intruders, causing information to be blocked, forged or forged, causing huge losses. Encryption algorithm is one of the effective measures to ensure information security, and encryption algorithm has good modification and portability. The research and application of encryption algorithms are very extensive. In the application of microcomputers, if the security of information transmission cannot be guaranteed, people should pay attention to it. Therefore, it is necessary to study the data encryption of single-chip computers and embedded systems, and the research in this area will be more extensive and
in-depth in the future.

References

[1] Cong L, Qiu F. Automotive Electronic Control System Unit based on Embedded Single Chip Microcomputer and CAN Bus[J]. Advances in Information Sciences & Service Sciences, 2013, 5(7):522-530.

[2] Pertuz S A, Llanos C, Pea C, et al. A parallel system-on-chip approach for impedance controller for a 7-DoF robotic hand[J]. Analog Integrated Circuits and Signal Processing, 2021, 106(1):195-204.

[3] Tragoudas S, Nagarandal V. On-chip embedding mechanisms for large sets of vectors for delay test[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2005, 24(3):488-497.

[4] Reis R, Lubaszewski M, Jess J A G. Design of Systems on a Chip: Design and Test || Embedded Core-based System-on-Chip Test Strategies[J]. 2006, 10.1007/0-387-32500-X(Chapter 11):223-238.

[5] Xu H, Tian J. Research on automatic correction method of spiral bevel gear error reversal based on single chip microcomputer[J]. Journal of Physics: Conference Series, 2021, 1748(6):062002 (6pp).

[6] Song S. Design and implementation of speed detection and control module based on single-chip microcomputer control[J]. Revista de la Facultad de Ingenieria, 2017, 32(14):461-466.

[7] Pelissier F, Chenini H, Berry F, et al. Embedded multi-processor system-on-programmable chip for smart camera pose estimation using nonlinear optimization methods[J]. Journal of Real-Time Image Processing, 2016, 12(4):1-17.

[8] Yulisdin, Mukhils, Erma, et al. Design System on Chip PreAmp Embedded on Electrocardiograph Based 0,35 CMOS Technology[J]. Journal of Physics: Conference Series, 2019, 1175(1):12095-12095.

[9] Nti R B, Ryoo K. A lightweight RSA-based System-on-a-Chip Design for Constrained Application[J]. International Journal of Innovative Technology and Exploring Engineering, 2019, 8(8S2):721-728.

[10] Venkata S H, Likita R D, Shameem S. Design and implementation of wearable single chip system for sensing, processing and transmitting human gesture and motion data[J]. International Journal of Engineering & Technology, 2018, 7(2.8):335.