*Journal of Information Hiding and Multimedia Signal Processing* ©2018 ISSN 2073-4212
**Ubiquitous International** Volume **9**, Number **5**, September **2018**

# A Data Hiding Algorithm by Combining Segment Address and EMD

Bailong Yang, Wenqiang Zhao*, Xinli Yin, Miao Wang
Xi'an Research Institute of High Technology, Xi'an, China
qqingnine@163.com

Received January, 2018; revised May, 2018

Abstract. *The data hiding scheme of Exploiting Modification Direction (EMD) has good embedding efficiency and high image quality. Its actual maximum embedding capacity is 1bpp. The embedding capacity of the improved algorithms have been improved, but also new distortion has been introduced. Actually, the data hiding ability of EMD is not fully exploited. To solve this problem, a new data hiding scheme by combining segment address and EMD is proposed. In the EMD algorithm, 4 binary bits m($m_4m_3m_2m_1$) are concealed in two pixel-pairs. For higher embedding capacity, 5 binary bits m'($m_5m_4m_3m_2m_1$) are read each time. If the decimal number m' is less than 23, conceal m' in two pixel-pairs. Otherwise, append a pixel pair and conceal m' in three pixel-pairs by using the mode "segment address + offset address". The embedding capacity is increased effectively by this way. Because there is no new distortion introduced, the visual quality of ours is as good as the EMD's. The experimental results show that the algorithm can improve the image hiding capacity while ensure the stego's visual quality.*
**Keywords:** Data Hiding, Exploiting Modification Direction, Segment Address, Embedding capacity

1. **Introduction.** With the rapid development of Internet technology, it has become a development trend that people send images, audio, video and personal information through the Internet [1]. However, personal sensitive data transmitted over the Internet are seriously threatened by the existence of illegal attacks such as fraud and copying [2]. Data encryption and data hiding are two different ways to protect sensitive data. The encrypted personal sensitive data is a string of meaningless characters that can easily be noticed by attackers [3]. Invisibility is an important feature of data hiding [4], which can protect sensitive data from being destroyed. Data hiding can be classified into two types, irreversible data hiding [5–7] and reversible data hiding [8–11]. Generally, the irreversible data hiding has greater embedding capacity.

The Least Significant Bit (LSB) algorithm [12] is one of the simplest hiding techniques in spatial domain, which replaces the least significant bit with a secret bit. Although the LSB method is simple, its embedding efficiency is not high. The data hiding method named Exploiting Modification Direction (EMD) proposed by Zhang et al. [6] has high embedding efficiency. The EMD method fully exploits the modification directions and each secret digit in a $(2n + 1)$-ary notational system can be concealed in a group of vector pixels containing $n$ pixels. Theoretically, the hiding capacity of the EMD method is $\log_2(2n+1)/n$ bpp. Practically, EMD achieves its maximum hiding capacity of 1 bpp when $n$ equals 2. The EMD scheme has high efficiency and many improved algorithms based on EMD have been proposed.

In 2008, Lee et al. [13] proposed an improved EMD algorithm. Although it greatly improved the embedding capacity, the algorithm's average peak signal-to-noise ratio (PSNR) decreased by 8dB compared with EMD. In 2009, Jung K H et al. [14] proposed an another improved EMD algorithm. This algorithm concealed one secret digit in the base-$(2n+1)$ notational system in a pixel, where $n = 2$. Obviously, the embedding capacity of Jung's is twice of Zhang's. However, the improved embedding capacity is at the cost of reducing the image quality. Kieu et al. [15] proposed the Fully Exploiting Modification Directions (FEMD) scheme in 2011. In this scheme, a secret digit in the base-$s^2$ notational system could be hidden into a pixel pair, and the embedding capacity is $\lfloor \log_2 s^2 \rfloor/2$ bpp in theory. Although Kieu et al. improve the embedding rate, the image quality decreases with the increase of the embedding rate. The average PSNR has been reduced to 31.69 dB when the embedding rate reaches 4.5bpp ($s = 23$). Kuo et al. [16] proposed the General EMD (GEMD) scheme in 2013. This method uses a flexible grouping strategy to embed the binary data stream and reduces the computational complexity well. In terms of embedding rate, this method also increases the embedding rate at the expense of image quality. In 2015, Lee et al. [17] proposed an adjustment hiding method based on EMD. In Lee's method, each secret digit in a $(c^n)$-ary notational system is carried by $n$ cover pixels. This method has various embedding rate and the visual quality is reduced with the increase of embedding rate. Qin et al. [3] proposed a reversible data hiding method based on EMD in 2015. Two stego images are produced after embedding process complete, and one degraded seriously. In addition, the overflow problem may occur by modifying the pixel value. Kuo et al. [18] combined FFEMD [19] with pixel difference and proposed an improved data hiding algorithm in 2016. The algorithm can conceal the secret data in any format according to the complex of adjacent pixels. At the same time, the overflow problem has been solved. The embedding rate is 1.5–1.69 bpp, and the image quality reduces by about 3–5dB compared with the EMD algorithm. In the EMD-based data hiding algorithm, the embedding rate decreases as the parameter $n$ increases. For solving this problem, Wang et al. [20] proposed a large capacity data hiding algorithm based on GEMD in 2017. The embedding rate remains 2bpp in Wangs scheme. The best visual quality (PSNR) is about 45.12 db.

The embedding rate is improved while the visual quality is decreased for most of the data hiding algorithm based on EMD. In order to improve the embedding rate without reducing the image quality, we propose a novel data hiding scheme by combining segment address and EMD. In the proposed scheme, the embedding capacity is about 9.62% higher than that of the EMD algorithm. The image quality of the proposed scheme is as good as that of EMD because no new distortion is introduced.

## 2. **The EMD scheme.**

### 2.1. **The algorithm principle.**

The main idea of the EMD method is that each secret digit $d$ in a $(2n+1)$-ary notational system is carried by $n$ cover pixels, and, at most, only one pixel is increased or decreased by 1 [6].

The EMD algorithm uses the extraction function $f$ to compute the weighted sum modulo $(2n+1)$ for each pixel group $(g_1 g_2,...,g_n)$, as shown in equation (1):

$$f(g_1, g_2, ..., g_n) = [\sum_{i=1}^{n} (g_i \cdot i)] mod(2n+1) \tag{1}$$

The value $f$ of the pixel group $(g_1 g_2,...,g_n)$ is easily worked out. Taking $n = 2$ as an example, Figure 1 shows the $f$ values for different pixel pair $(g_1,\ g_2)$. In Figure 1, the

abscissa represents $g_1$, the ordinate represents $g_2$, and the $f$ value of any pixel pair ($g_1$, $g_2$) can be easily calculated.

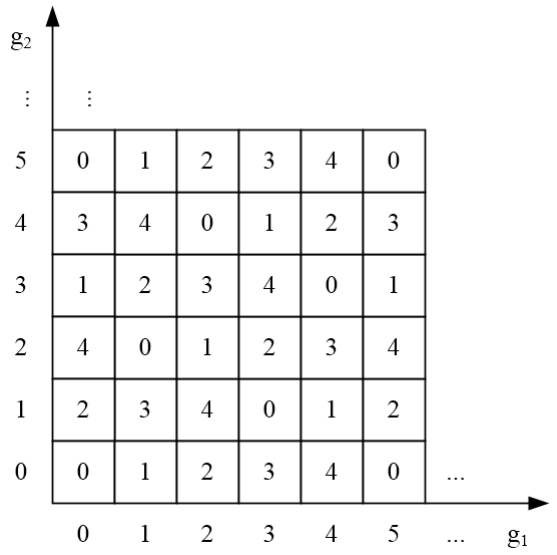


FIGURE 1. The value of $f$

Each pixel group ($g_1 g_2,\ldots,g_n$) can conceal a secret data $d$ in $(2n + 1)$-ary notational system. If $d = f$, no pixel in this pixel group does not need to be modified. If $d \neq f$, calculate $s$ according to formula (2)

$$s = (d - f) mod (2n + 1) \tag{2}$$

If $s$ is no more than $n$, increase the value of $g_s$ by 1, otherwise, decrease the value of $g_{2n+1-s}$ by 1. For example, there is a pixel group ($g_1$, $g_2$, $g_3$, $g_4$) = (137,139,141,140) with $n = 4$, and the secret data $d = (100)2 = (4)9$. The embedding procedure as following:

Step 1. Compute $f = (g_1 \times 1 + g_2 \times 2 + g_3 \times 3 + g_4 \times 4)\ mod\ (2n{+}1)$
$= (137{\times}1{+}139{\times}2{+}141{\times}3{+}140{\times}4)\ mod\ 9$
$= 3.$

Step 2. Compute $s = (d{-}\mathrm{f})\ mod\ (2n{+}1)$
$= (4{-}3)\ mod\ 9$
$= 1.$

Step 3. Because $s$ is no more than $n$, increase the value of $g_s$ by 1. So, it's easy to get $g_s$'= 138.

The stego pixel group ($g_1$', $g_2$', $g_3$', $g_4$') = (138,139,141,140). In another case, if the secret data $d = (0)2 = (0)9$, easy to get $s = 6$. Decrease the gray value of $g_{2n+1-s}$ by 1, and the stego pixel group is (137,139,140,140). On the receiving side, the secret digit can be easily extracted by calculating the extraction function of stego pixel group.

Embedding secret information may cause overflow problems. For this problem, please read the paper [6], no longer described here.

2.2. **Embedding capacity.** Theoretically, the embedding capacity of EMD is $\log_2(2n{+}1)/n$ and the maximum is 1.1610bpp with $n = 2$. Actually, the maximum embedding rate is 1bpp when $n = 2$.

2.3. **Analysis.** Assume $n = 2$, four secret bits in the binary stream are read each time and converted into two digits in a 5-ary notational system. One digit is hidden into a pixel pair, and another digit is hidden into the following pixel pair. It is obvious that the

actual embedding rate is 1bpp. The range of two digits in a 5-ary notational system and 4 binary bits in decimal number is 0–24 and 0–15, respectively. It can be seen that 16–24 of the two digits have never been used, the ability to hide data is not fully exploited.

The same problem exists when $n$ takes other values.

3. **Segment address.** 8086 CPU is a 16-bit structure that is handled, transmitted, and temporarily stored in 16-bit locations internally. But its address bus is 20 bits and its addressing capability is 1M [21]. In order to reach the 1MB addressing capability, 8086 CPU uses a method of synthesizing two 16-bit addresses internally to form a 20-bit physical address [22]. That is, when the CPU accesses memory, the physical address of the memory unit is obtained by adding a base address (segment address × 16) and an offset address. The principle is shown in Figure 2.

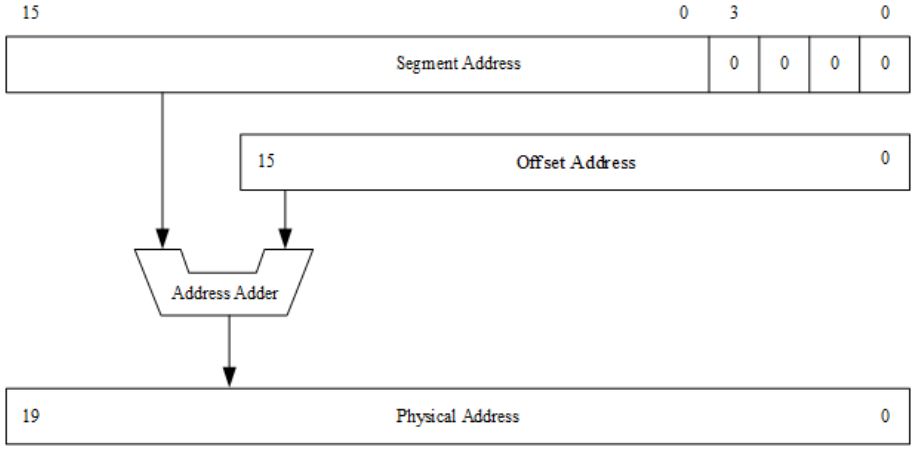


FIGURE 2. Addressing mode

4. **Proposed method.** The embedding capacity of EMD is $\log_2(2n+1)/n$ bpp in theory, and the actual embedding rate is smaller than this value. The reason why the theoretical value is different from the actual value is that the ability to hide information is not fully exploited. When $n = 2$, the embedding capacity of EMD is the largest. In the following section, we mainly discuss the case with $n = 2$.

In the previous section, we discuss the shortcomings of the EMD algorithm. In order to make full use of the pixel pair's capability of data hiding, this paper propose a new scheme that hiding a secret digit m' contains 5 bits into two pixel pairs $S$ each time. However, 0 ≤ m' ≤ 31, it is beyond the ability of the two pixel pairs to hide m'. This problem can be solved by using the idea of "physical address = segment address × 16 + offset address ".

In the proposed method, five secret bits $m' = (m_1 m_2 m_3 m_4 m_5)$ are read at a time. When m' ≤ 22, two pixel pairs can satisfy the requirement of hiding m'. When m' ≥ 23, a pixel pair is added after these pixel pairs, and the secret information is hidden by the method of "segment address + offset address".

4.1. **The process of data hiding.** Read 5 bits $(m_1 m_2 m_3 m_4 m_5)$ from the secret binary stream, and convert them into decimal number $m'$.

If $m' \leq 22$

Convert $m'$ into digits in a 5-ary notational system,

and then embed them into two pixel pairs.

elseif $23 \leq m' \leq 27$

Add a pixel pair $b_1$ behind 2 pixel pairs $b_2$.

Embed the decimal number 23 into $b_2$,
and embed the decimal number $k$ into $b_1$, where $m$' $= 23 + k$.
elseif $28 \leq m$' $\leq 31$
Add a pixel pair $b_1$ behind 2 pixel pairs $b_2$.
Embed the decimal number 28 into $b_2$,
and embed the decimal number $k$ into $b_1$, where $m$' $= 28 + k$.
end
Repeat the above steps until all the secret information is embedded.

4.2. **The process of data extraction.** Read 2 pixel pairs in sequence, one digit in a 5-ary notational can be produced by each pixel pair. Then, the decimal number $m$' is produced by the 2 digits.
If $m$' $\leq 22$
Convert $m$' to binary stream, which contains 5 bits.
elseif $m$' $= 23$
Read a pixel pair more, calculate this pixel pair's value $f$ according to formula (1).
The secret digit in decimal number is $m$' $= 23 + f$, and then convert $m$' into binary stream.
elseif $m$' $= 24$
Read a pixel pair more, calculate this pixel pair's value $f$ according to formula (1).
The secret digit in decimal number is $m$' $= 28 + f$, and then convert $m$' into binary stream.
end

4.3. **Theoretical embedding capacity.** Assume that the length of secret binary stream is $5n$. 5 bits are read each time and then converted into decimal number $m$'. These information bits can be hidden into 4 pixels when $m$' is less than 23, otherwise 6 pixels are needed. The probability that $m$' is less than 23 is 23/32, and another case is 9/32.
The embedding capacity of proposed method is

$$\frac{5n}{n \times 4 \times \frac{23}{32} + n \times 6 \times \frac{9}{32}} = 1.0959bpp$$

Because no new distortion is introduced in the embedding process, the embedding efficiency and the stego image quality in this paper is as same as that of the EMD algorithm.

4.4. **When $n$ takes other values.** In the previous section, we have analyzed the principle of the EMD scheme and give an improved method when $n = 2$. When $n$ takes other values, a similar approach can be used to improve the embedding capacity. In the process of data hiding, the data embedding can be completed by reference to the analysis method of $n = 2$.

5. **Experiments.** To evaluate the performance of the proposed method, we implemented the proposed method and the EMD method in MATLAB2013A. In this experiment, six gray-scale images sized 256×256 named Baboon, Boat, F16, Goldhill, Lena, Pigeon are used as the test image, shown in Figure 3. The secret binary stream is generated by a random function.
In the experiment, the proposed method is tested in terms of embedding capacity and visual quality of stego images. The embedding capacity (EC) is the total number that the secret binary could be hidden into an image. The visual quality is measured by the peak-signal-to-noise ratio (PSNR).

The EMD algorithm achieves its maximum embedding capacity with $n = 2$. The experimental section will distinguish the performance of the algorithm when $n = 2$ and $n$ takes other values.

(a)Baboon (b)Boat (c)F16 (d)Baboon (e)Baboon (f)Pigeon

FIGURE 3. Test images

5.1. **The performance of the algorithm when $n = 2$.** The embedding capacity (EC) and the visual image quality (PSNR) is shown in Table 1, where $n = 2$. It is clear that the embedding capacity of proposed method is larger than the EMD's. The average embedding capacity is increased by 9.62%. This is due to the ability for pixel pair to hide information is fully exploited in proposed method. In addition, it is known from the table that our scheme and EMD method have same visual quality. This illustrates that the proposed method increases the embedding capacity without introducing new distortion. The experimental results are in agreement with the theoretical expectations.

The embedding efficiency of the proposed method is the same as the EMD method. For more about the performance of the algorithm, please read the literature [6].

TABLE 1. Performance results of EMD method and the proposed method with $n = 2$

| Test image | EMD method | | proposed method | |
|---|---|---|---|---|
| | PSNR | EC | PSNR | EC |
| Baboon | 52.1191 | 65536 | 52.1189 | 71765 |
| Boat | 52.1032 | 65536 | 52.1032 | 71895 |
| F16 | 52.1119 | 65536 | 52.1147 | 71805 |
| Goldhill | 52.1264 | 65536 | 52.1259 | 71795 |
| Lena | 52.1129 | 65536 | 52.1118 | 71875 |
| Pigeon | 52.0984 | 65536 | 52.0957 | 71905 |
| average | 52.1120 | 65536 | 52.1110 | 71840 |

5.2. **The performance of the algorithm when $n$ takes other values.** In this section, the performance is analyzed when $n$ takes other values, and the two aspects of embedding rate (ER) and image visual quality are mainly considered. The experimental results are shown in Figure 4 and Figure 5, respectively. It is still effective when $n$ takes other values. We can see from Figure 4 that the proposed method has a higher embedding capacity compared with EMD, where $n$ takes different values. In Figure 5, it is obvious that the two method has the same image visual quality when they have same parameter $n$. This shows that the proposed method improves the embedding capacity, but no new distortion is introduced.

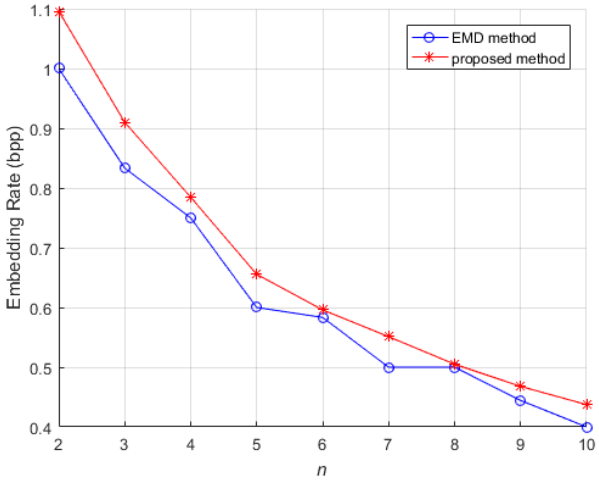


FIGURE 4. Performance results of the embedding rate (ER) for various values of $n$.

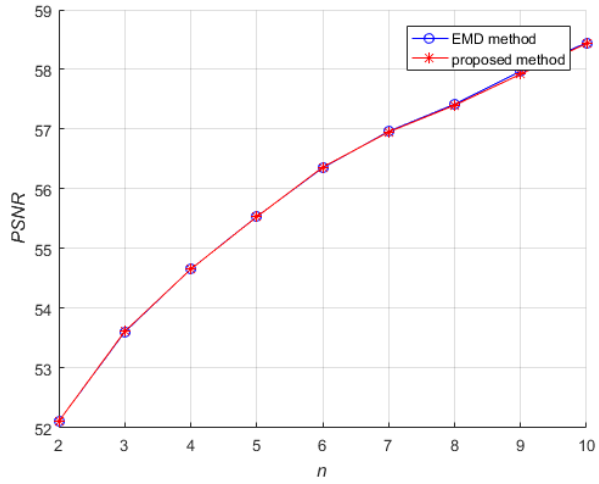


FIGURE 5. Performance results of the image visual quality (PSNR) for various values of $n$.

6. **Conclusions.** In this paper, we analyze the principle of EMD and point that the embedding capacity could be improved by fully exploiting the ability for pixel pair to hide information. To solve this problem, a data hiding algorithm combining segment address and EMD was proposed. The proposed algorithm makes full use of the ability for pixel pair to hide information, and the average embedding capacity is increased by about 9.62% (for example, $n = 2$). The visual image quality of the proposed algorithm is as the same as that of EMD because no new distortion is introduced. There is a consistency between experimental results and theoretical expectations, which indicates that the proposed algorithm is feasible.

## References

[1] S.A. Parah, J.A. Sheikh, N.A. Loan, et al, A Robust and Computationally Efficient Digital Watermarking Technique Using Inter Block Pixel Differencing, *Springer International Publishing*, 2017.

[2] G. Badshah, S.C. Liew, J.M. Zain, et al, Watermark Compression in Medical Image Watermarking Using Lempel-Ziv-Welch (LZW) Lossless Compression Technique, *Journal of Digital Imaging*, vol.29, no.2, 2016.

[3] C. Qin, C.C. Chang, T.J. Hsu, Reversible data hiding scheme based on exploiting modification direction with two steganographic images, *Multimedia Tools and Applications*, vol.74, no.15, pp.5861–5872, 2015.
[4] Q.L. WU, M. WU, Novel Audio Information Hiding Algorithm Based on Wavelet Transform, *Journal of Electronics & Information Technology*, vol.38, no.4, pp.834–840, 2016.
[5] J. Mielikainen, LSB matching revisited, *IEEE Signal Processing Letters*, vol.13, no.5, pp.285–287, 2006.
[6] X. Zhang, S. Wang, Efficient Steganographic Embedding by Exploiting Modification Direction, *Communications Letters IEEE*, vol.10, no.11, pp.781–783, 2006.
[7] W.C. Kuo, M.C. Kao, A Steganographic Scheme Based on Formula Fully Exploiting Modification Directions, *Ieice Transactions on Fundamentals of Electronics Communications & Computer Sciences*, vol.96, no.11, pp.2235–2243, 2013.
[8] A.M. Alattar, Reversible watermark using the difference expansion of a generalized integer transform, *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, vol.13, no.8, pp.1147–1156, 2004.
[9] S. Weng, Y. Zhao, J.S. Pan, et al, Reversible Watermarking Based on Invariability and Adjustment on Pixel Pairs, *IEEE Signal Processing Letters*, vol.15, no.20, pp.721–724, 2008.
[10] S. Weng, J.S. Pan, L. Li, et al, Reversible data hiding based on an adaptive pixel-embedding strategy and two-layer embedding, *Information Sciences*, no.369, pp.144–159, 2016.
[11] S. Weng, Y. Liu, J.S. Pan, et al, Reversible data hiding based on flexible block-partition and adaptive block-modification strategy, *Journal of Visual Communication & Image Representation*, no.41, 2016.
[12] L.F. Turner, *DIGITAL DATA SECURITY SYSTEM*, 1991.
[13] C.F. Lee, C.C. Chang, K.H. Wang, An improvement of EMD embedding method for large payloads by pixel segmentation strategy, *Butterworth-Heinemann*, pp.1670–1676, 2008.
[14] K.H. Jung, K.Y. Yoo, Improved Exploiting Modification Direction Method by Modulus Operation, *International Journal of Signal Processing Image Processing & Pattern Recognition*, vol.2, no.1, pp.79–88, 2009.
[15] T.D. Kieu, C.C. Chang, A steganographic scheme by fully exploiting modification directions, *Expert Systems with Applications An International Journal*, vol.38, no.8, pp.10648-10657, 2011.
[16] W.C. Kuo, L.C. Wuu, S.H. Kuo, The high embedding steganographic method based on general multi-EMD, *International Conference on Information Security and Intelligence Control*, pp.286–289, 2013.
[17] C.F. Lee, C.C. Chang, P.Y. Pai, et al, Adjustment hiding method based on exploiting modification direction, *International Journal of Network Security*, vol.17, no.5, pp.607–618 2015.
[18] W.C. Kuo, J.J. Li, C.C. Wang, et al, An Improvement Data Hiding Scheme Based on Formula Fully Exploiting Modification Directions and Pixel Value Differencing Method, *Asia Joint Conference on Information Security*, pp.136–140, 2016.
[19] S.Y. Shen, L.H. Huang, A data hiding scheme using pixel value differencing and improving exploiting modification directions, *Computers & Security*, vo.48, pp.131–141, 2015.
[20] C.C. Wang, W.C. Kuo, Y.C. Huang, et al, A high capacity data hiding scheme based on re-adjusted GEMD, *Multimedia Tools & Applications*, vol.8, pp.1–15, 2017.
[21] B.B. Brey, Intel microprocessor: from 8086 to Pentium series architecture, programming and interface technology, *Higher Education Press*, 2001.
[22] S. Wang, Assembly language, *Tsinghua University press*, 2003.