S. Tynymbayev¹, Y. Zh. Aitkhozhayeva², S. Adilbekkyzy³

¹Institute of Information and Computational Technologies, Almaty, Kazakhstan;
²Kazakh National Research Technical University named after K. I. Satpayev
(Satbayev University), Almaty, Kazakhstan;
³Universiti Tenaga Nasional (UNITEN), Kajang, Selangor, Malaysia.
E-mail: s.tynym@mail.ru, ait_djam@mail.ru, sairan02.95@mail.ru

HIGH SPEED DEVICE FOR MODULAR REDUCTION

Abstract. It points to the advantages of hardware implementation of encryption. Hardware implementation of cryptosystems allows to increase their speed. But the low-speed of asymmetric cryptosystems, in comparison with symmetric cryptosystems, even with hardware implementation limits their application. The most used asymmetric crypto algorithm is the RSA encryption algorithm. Modular reduction is a time-critical operation that slows down the implementation of the RSA algorithm. The structure of a fast modular reduction device is proposed, in which a modified division method with a shift of the remainders by two bit positions to the left is used. This allows to speed up the receipt of the remainder twice.

Keywords: hardware encryption, asymmetric cryptoalgorithms, modular reduction.

Introduction. Cryptographic methods of information protection are indispensable in the transfer of confidential information through communication channels, establishing the authenticity of transmitted messages, storing information on storage media. Hardware implementation of cryptographic algorithms allows you to perform data encryption much faster and safer than software implementation. The development of modern microelectronics allows you to place the cryptoprocessor on a single chip [1-3].

Specialized hardware devices of cryptographic protection are not only more reliable and productive in comparison with software encryption. The list of advantages of hardware encoders that realize both symmetric and asymmetric crypto algorithms is much wider [4].

In practical implementation, the problem of high-speed symmetric cryptosystems (systems with a secret key) is the problem of key distribution. To solve this problem, asymmetric cryptosystems (two-key systems with a public key) were proposed. Asymmetric cryptosystems, in comparison with symmetric cryptosystems, have a lower speed, but there is no problem of transferring and confirming the authenticity of secret keys. Cryptography with public keys is better corresponds for key management and for protocols.

Algorithm of RSA encryption. From asymmetric cryptoalgorithms, the RSA encryption algorithm (Rivest, Shamir and Adleman, 1978), which is based on an irreversible transformation (decomposition of large numbers into prime factors), is most widely used in practice. RSA is part of ISO 9796, it is used as a public key encryption standard in the banking sector of France and Austria. Currently, the RSA algorithm is used in many protocols and programs, including:

- the S/MIME (Secure/Multipurpose Internet Mail Extensions) application layer protocol for encrypting and signing in e-mail using a public key;
- the SSH application layer protocol, in which the algorithms for the digital signature of RSA (DSA) is used for server authentication;
- the TLS (Transport Layer Security) presentation layer protocol and its predecessor SSL (Secure Sockets Layer), which are the basis of HTTPS (Hyper Text Transfer Protocol Secure);
- a set of IPSec (IP Security) network layer protocols, including authentication, integrity check and encryption of IP packets;
the STT (Stateless Transport Tunneling) tunneling protocol for network virtualization;
- the PGP (Pretty Good Privacy) program that allows you to perform encryption and digital signature operations for messages, files and other information presented in electronic form, including transparent data encryption on storage devices, for example, on a hard disk;
- a family of standards PKCS (Public Key Cryptography Standards) designed for secure information exchange on the Internet using PKI (Public Key Infrastructure).

Therefore, many studies are focused at improving the performance of crypto-algorithm RSA.

**Hardware solutions for modular reduction.** To develop a high-speed RSA cryptoprocessor, it is necessary to develop fast-acting blocks of hardware implementation of algorithm operations. The basic operation of the RSA algorithm is the modular exponentiation of integers \((a \mod p)\). This operation is realized through multiplication, squaring and modular reduction. One of the approaches to improve the performance of public key cryptosystems is the acceleration of these operations. The most complex of them is the modular reduction operation, since it is the calculation of the remainder from dividing the number by the module \(P\), and the division operation is the most complex of the arithmetic operations.

Theoretical and practical questions of high-speed integer multipliers and quadrants for a different class of computing systems are well developed, which cannot be said about the modular reduction. The high-speed hardware solution of the modular reduction operation is a key problem in the hardware implementation of asymmetric cryptoalgorithms that use the modular exponentiation of numbers, including RSA.

There are many different methods of calculating the remainder when dividing by the module \(P\) [5-10]. When using the binary (usual) representation of integers, it is possible to distinguish three types of device structures of modular reduction depending on the principle of remainder formation.

In the first type of devices blocks of formation of multiple modules \(P^*i (i = 1, 3, ..., k)\) are used. Then these values are simultaneously (in parallel) subtracted from the reducible number \(A\) on \(K\) adders. The least positive remainder \(C_1=A-P^*i\) is the result [11]. This type of device has a high speed, but with increasing values of \(A\) and \(P\), the complexity of circuits and hardware costs increase. RSA uses numbers of the order of \(10^{30}\), which makes it impossible to use this type of device in the practical implementation of RSA.

The second type of devices uses the method of forming the remainders \((r_i)\) of the bit weights of the binary number \((2^i)\) from division by module \(P\). The calculated remainders modulo \(2^i (i = 0, 1, ..., k-1)\) are summed if the coefficients of the corresponding weights of the number \(A_i\) are equal to 1. The summation is carried out successively on \(K-1\) modulo adders \(P\) [12]. It is implemented by the formula: \(A \mod P = (\sum_{0}^{k-1} (2^i \mod P)A_i) \mod P\). Sequential summation on \(K-1\) adders for large digits \((k-1)\) has a negative effect on the speed of the device.

The third type of devices uses various modified methods of a machine algorithm for binary division, which leads to a wide variety of structures. When division is used with a divisor shift to the right, it is possible to obtain various device structures, one of which is given in [13].

**Device structure for modular reduction.** In this paper, we consider a device for fast modular reduction of a number with a shift of the remainder to the left. At each step of the calculation, the value of either tripled \((3p)\) or doubled \((2p)\) or single value of the module \((p)\) is subtracted from the remainder \(r_{i+1}\) that shifted by two bit positions to the left. This allows to accelerate the calculation of reducing the 2\(n\)-bit number \(A\) by the \(n\)-bit module \(P\) twice.

The binary representation \((2p\) and \(3p)\) and ones' complement \((2\bar{p}\) and \(3\bar{p})\) of the doubled and tripled module are precomputed. Then, previous remainder \(r_{i+1}\) multiplied by the four, i.e. \(4r_{i+1}\) is compared on the comparators with the values \(3p, 2p, p\) and it is determined which of the following operations must be performed in order to compute the value of the next remainder \(r_i\): \(r_i = 4r_{i+1} - p\), or \(r_i = 4r_{i+1} - 2p\), or \(r_i = 4r_{i+1} - 3p\).

In the adder all operations are performed in the two's complement, so \(r_i = 4r_{i+1} + \bar{p} + 1\), or \(r_i = 4r_{i+1} + 2\bar{p} + 1\) or \(r_i = 4r_{i+1} + 3\bar{p} + l\) is determined and formed.

A functional diagram of such a device is shown in figure 1.

The device consists of a \((2n+2)\)-bit register \(RGP\), where the reducible number \(A\) is stored and shifted by two bit positions to the left, the register \(RGP\) for storing the \(n\)-bit module \(P\), the adder \(Addl\) for calculating the ones' complement of the tripled module value \(3\bar{p}\) (by summing \(2\bar{p}\) with \(p\)).
Values of $3p$ are formed by inverting the output bits of $Add1$ on the block of inverters $Inv1$. In the comparator $Com1$, the codes $4r_{r,i}$ and $p$ are compared.

If $4r_{r,i} < p$, then a "0" signal is generated at its output. Conversely, if $4r_{r,i} \geq p$, then the signal "1" is generated at the output of this circuit.

The $Com2$ compares the value $4r_{r,i}$ with the value of the module $2p$. If $4r_{r,i} < 2p$, then at the output 1 of this circuit, the signal "1" is set. If $4r_{r,i} \geq 2p$, output 1 is set to "0" and at the output 2 is signal "1".

The $Com3$ compares the codes $4r_{r,i}$ and $3p$. If $4r_{r,i} < 3p$, then at the output 1 of this circuit a "1" signal is formed and "0" is set at the output 2. In case, $4r_{r,i} \geq 3p$, at the output 1 is formed by the signal "0" and at the output 2 the signal "1" is set.

Table 1 shows the executable operations, depending on the ratios of $4r_{r,i}$ with different values of the modules $p$, $2p$ and $3p$.

<table>
<thead>
<tr>
<th>Ratios</th>
<th>Executable operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>$4r_{r,i} &lt; p$</td>
<td>$r_i = 4r_{r,i}$</td>
</tr>
<tr>
<td>$p \leq 4r_{r,i} &lt; 2p$</td>
<td>$r_i = 4r_{r,i} + \frac{p}{2}$</td>
</tr>
<tr>
<td>$2p \leq 4r_{r,i} &lt; 3p$</td>
<td>$r_i = 4r_{r,i} + \frac{3p}{2}$</td>
</tr>
<tr>
<td>$3p \leq 4r_{r,i}$</td>
<td>$r_i = 4r_{r,i} + 3p$</td>
</tr>
</tbody>
</table>

According to this table, when $4r_{r,i} < p$, the value $4r_{r,i}$ with $OR1$ gates is written without changes to $RgA$.

With the ratios $p \leq 4r_{r,i} < 2p$, a signal "1" is generated at the output of the $AND6$ gates, which is simultaneously fed to the inputs of $OR3$ and $AND8$ gates, the second input of the $AND8$ gates are supplied
with bits $\overline{p}$. Output AND8 gates are fed to the right inputs of the adder Add2 via the OR2 gates. On the left inputs Add2, the codes of the value $4r_{t,1}$ are fed, and through OR3 the signal "+1" is fed to the input of the lowest order bit position of this adder, the operation $r_{l} = 4r_{t,1} + \overline{p} + 1$ is performed. The result through the block of OR1 gates is transmitted to the highest order bit positions of the register $R_{g}A$.

When conditions $4r_{t,1} \geq 2p$ and $4r_{t,1} < 3p$ are satisfied, a signal "1" is generated at the output of the AND7 gates, which is fed to the input of the OR3 gate and the block of the AND9 gates. At the second data inputs of AND9 are fed the bits of module $2p$. Output AND9 gates through the block of OR2 gates are fed to the right inputs of the Add2, and the code "+1" is supplied to the input of the lowest order bit position and the operation $r_{l} = 4r_{t,1} + 2\overline{p} + 1$ is performed in the adder. The result through the block of OR1 gates is transmitted to the highest order bit positions of the register $R_{g}A$.

With the ratios $4r_{t,1} \geq 3p$ from output 2 of the comparator Com3, a signal "1" is applied to the input of the OR3 gate and to the control inputs of the AND10 gates. At the data inputs of AND10 gates are fed with bits of the module $p$ multiplied by three $(3p)$ from the outputs of the adder Add1. Codes $3\overline{p}$ through the block of OR2 gates are transmitted to the right inputs of the Add2, to the left inputs of this adder bits of code $4r_{t,1}$ are fed. In this case, the operation $4r_{t,1} + 3\overline{p} + 1$ is performed in the adder. The result of the operation through the block of OR1 gates is written to the highest order bit positions of the register $R_{g}A$.

The high-speed modular reduction device works as follows.

With the signal "Start", the reducible number $A$ and the module $P$ by means of the blocks AND3 and AND4 gates, respectively, are received in the registers $R_{g}A$ and $R_{g}P$. From the true outputs of the register the value of the true representation of the module $p$ is transferred to the right inputs of the comparator Com1 and shifted to the left by one bit $(2p)$ is fed to the right inputs of the comparator Com2. From the complementary outputs of $R_{g}P$, a ones' complement module $\overline{p}$, which is fed to the data inputs of the block of AND8 gates and to the left inputs of the adder Add1. The value $3\overline{p}$ from output Add1 is fed to the left inputs of AND10 gates. The value $3\overline{p}$ is inverted by the inverters block Inv1, forming the value $3p$, which is fed to the right inputs of the comparator Com3.

Also, with the signal "Start" through the block of AND2 gates the binary code of the number of cycles $K = n/2$ is received in the subtracting counter of the clock pulses (CCP). In addition, the "Start" signal, the delayed on delay line $DL1$ for the time of recording information in $R_{g}A$ and $R_{g}P$, is fed to the one-input of the flip-flop $T$. Flip-flop is set to the one condition. The one condition of the trigger permits the passage of the first clock pulse $CP1$ to the output of the $AND1$ gate. Further, $CP1$ arrives at the input of the shift register $R_{g}A$ and shifts it two bit positions to the left, increasing the contents of $R_{g}A$ by four. At the same time, the value of the counter decreases by one by the pulse $CP1$. During shift of information in $R_{g}A$ $CP1$ is delayed on $DL2$. After this, the shifted by two bit positions content of $R_{g}A$ through the block of the $AND5$ gates is transferred to the left inputs of the adder Add2, Com3, Com2, Com1.

Further, depending on the ratio of the $4r_{t,1}$ code and the values of the modules $3p$, $2p$ and $p$, a signal "1" is generated either at the output of the AND6 or AND7 gates, or at the output 2 of the comparator Com3. According to the generated signal "1", the calculation $r_{l}$ is performed with reference to table 1. The resulting partial remainder $r_{l}$ by the block OR1 gates is transmitted to the register $R_{g}A$, being memorized in the highest order bit positions of the register $R_{g}A$. At this point, the circuit receives a second clock pulse $CP2$, which passes through the block of AND1 gate and shifts $R_{g}A$ another two bit positions to the left, forming the value $4r_{l}$. Simultaneously, $CP2$ arrives at the subtracting input of the CCP and reduces its state by one. The value $4r_{l}$ from the outputs of the AND5 gates goes to the inputs Add2, Com3, Com2, Com1. In the adder Add2 the intermediate remainder $r_{l-1}$ is calculated, which is transmitted to the highest order bit positions of the register $R_{g}A$.

After the $n/2$th clock pulse arrives in the CCP, a zero code is set and the signal "End of operation" is generated, which is fed to the zero-input of the flip-flop $T$ and blocks the passage of the next clock pulse to the output of the AND1 gate. The last clock pulse calculates the last remainder $r_{n/2}$, which is stored in the highest order n-bit positions of the register $R_{g}A$, which is the result of the calculation. The result from $R_{g}A$, on the signal "End of Operation", which was delayed on $DL3$, is output by the block the $AND11$ gates to the output.
Results. The presented device allows to accelerate the calculation by reducing the 2n-bit number \( A \) modulo \( P \) by two times. The number of cycles necessary to reducing any number \( A \) modulo \( P \) is defined as \( K = \frac{n}{2} \), where \( n \) is bits of the module \( P \). For this device, a certificate of authorship has been obtained [14].

Conclusion. When processing a large amount of data on the same algorithm, the most productive are the pipeline structures. When encrypting data, the modular reduction operation is performed for a large amount of different numbers. Therefore, to increase the speed, it is advisable to use pipeline structures. When pipelining, the whole process is divided into a sequence of completed steps. Each of the stages of the division procedure is computed at its stage pipeline, with all stages running in parallel.

On the base of the modification of the above device, it is possible to construct a pipelined device for formation of the remainders by arbitrary module \( P \) of the number \( A \).

REFERENCES


С. Т. Тыныбаяев1, Е. Ж. Айтхозаева2, С. Эдилбеккызы3

1Академия наук Республики Казахстан,
2К. И. Сатпаев атындағы Қазақ ұлттық техникалық зерттеу университеті, Алматы, Казахстан,
3Universiti Tenaga Nasional (UNITEN), Kajang, Selangor, Malaysia

ЖЫЛДАМДЫҒЫ ЖОГАРЫ МОДУЛЬГЕ КЕЛТІРУ ҚУРЫЛГЫСЫ

Аннотация. Криптографиялық аппаратты жолмен іске асыру олардың жылдамдығын арттыруға мүмкіндік береді. Алайде асимметриялық криптоалгоритмдердің төмен жылдамдығы олардың қолданылуын шектейді. Қолданыска қарсы асимметриялық криптоалгоритм RSA шифрлау алгоритмі болып табылады. Модульге келтіру операциялар ішінде RSA алгоритмі іске асырылуы бұғылы ұакыты өңірдің қысылығын болып табылады. Қалықтық екі разрядқа солға жылжытатының бөліу әдісінің туралық қолданылыңың жылдамдығы жогары модульге келтіру құралының құрылысы ұсыналды. Бұл қалықтық алуы екі сөсеге жылдамдатуға мүмкіндік береді.

Түйін сөздел: аппаратты шифрлау, асимметриялық криптоалгоритмдер, модульге келтіру.

С. Т. Тыныбаяев1, Е. Ж. Айтхозаева2, С. Эдилбеккызы3

1Институт информационных и вычислительных технологий, Алматы, Казахстан,
2Қазақский национальный исследовательский технический университет им. К. И. Сатпаева, Алматы, Казахстан,
3Universiti Tenaga Nasional (UNITEN), Kajang, Selangor, Malaysia

БЫСТРОДЕЙСТВУЮЩЕЕ УСТРОЙСТВО ДЛЯ ПРИВЕДЕНИЯ ЧИСЕЛ ПО МОДУЛЮ

Аннотация. Аппаратная реализация криптосистем позволяет повысить их быстродействие. Но низкое быстродействие асимметричных криптосистем ограничивает их применение. Самым используемым асимметричным криптоалгоритмом является алгоритм шифрования RSA. Приведение по модулю является критичной по времени операцией, замедляющей реализацию алгоритма RSA. Предлагается структура устройства быстрого приведения по модулю, в котором используется модифицированный метод деления со сдвигом остатков на два разряда влево. Это позволяет ускорить получение остатка в два раза.

Ключевые слова: аппаратное шифрование, асимметричные криптоалгоритмы, приведение по модулю.

Information about authors:
Tynymbayev Sakilybay – leading researcher, Candidate of Technical Sciences, Institute of Information and Computational Technologies, Almaty, Kazakhstan; s.tynym@zmail.ru; https://orcid.org/0000-0002-9326-9476
Aitkhozhayeva Yevegniya – associated professor of the Department of Information Security, Candidate of Technical Sciences, Kazakh National Research Technical University named after K. I. Satpayev, Almaty, Kazakhstan; ait_djam@mail.ru; https://orcid.org/0000-0002-5961-8556
Adilbekkyzy Sairan – master’s degree student Universiti Tenaga Nasional, Kajang, Selangor, Malaysia, Almaty, Kazakhstan; sairan.02.95@mail.ru; https://orcid.org/0000-0002-3929-7070