УДК 004.4; 004.9

Каленик А.Н., Коляда А.А., Мазуренко П.А., Шабинская Е.В.

АЛГОРИТМ УМНОЖЕНИЯ ПО БОЛЬШОМУ МОДУЛЮ НА ОСНОВЕ ОПТИМИЗИРОВАННОЙ МОДУЛЯРНОЙ СХЕМЫ МОНТГОМПРИ С ИСПОЛЬЗОВАНИЕМ МИНИМАЛЬНОГО КОМПЛЕКТА ТАБЛИЦ

Представлен новый алгоритм умножения по большому модулю, базирующийся на оптимизированной минимально избыточной модулярной схеме Монтгомери и имеющий таблично-сумматорную конфигурацию. Обеспечивается высокое быстродействие при использовании минимального комплекта таблиц.

В современном процессе развития средств защиты информации важное место отводится проблематике разработки высокопроизводительных вычислительных технологий (ВТ) на диапазонах больших чисел. К таким технологиям относится, в частности, модулярная ВТ. В настоящее время на ее основе создано целое семейство быстрых алгоритмов умножения и возведения в степень по большим модулям [1−7]. Благодаря кодовому параллелизму модулярных систем счисления (МСС) применение алгоритмов данного класса обеспечивает существенное повышение скорости выполнения криптографических преобразований. Ниже приводится описание алгоритма умножения по большому модулю p, базирующегося на оптимизированной минимально избыточной модулярной схеме Монтгомери [4 − 6] и имеющего таблично-сумматорную конфигурацию. Обладая всеми важнейшими реализационными свойствами, присущими модулярным вычислительным структурам, синтезированный алгоритм позволяет достичь высокого быстродействия при использовании минимального комплекта таблиц (КТ).

Введем обозначения:

$\mathbf{Z}_m = \{0, 1, \ldots, m-1\}$ (m – натуральный модуль);

$|x|_m$ – элемент множества $\mathbf{Z}_m$, сравнимый с величиной x (в общем случае рациональным числом) по модулю m;

$(\chi_1, \chi_2, \ldots, \chi_l)$ – код целого числа (ЦЧ) X в МСС с базисом $\mathbf{M}_1 = \{m_1, m_2, \ldots, m_l\}$ ($\chi_i = |X|_{m_i}$ $(i = \overline{1, l})$, m_i – простое число).

Алгоритм умножения по модулю *p* на основе минимального КТ

Параметры алгоритма: основания – простые числа m_1, m_2, ..., m_l, m_{l+1}, m_{l+2}, …, m_k МСС ($m_k \geq 2m_0 + \_l - 2,\ m_0 \geq \_l - 2,\ \_l = k - l,\ 1 < l < k,\ k \geq 2$) и модуль $p = (\pi_1, \pi_2, \ldots, \pi_l, \pi_{l+1}, \pi_{l+2}, \ldots, \pi_k)$ ($\pi_i = |p|_{m_i}$ $(i = \overline{1, k})$), а также разрядности b_0 и b_1 ($2^{b\_0} \leq \min\{m_1, m_2, \ldots, m_k\}$; $b\_0 + b\_1 \leq 32$ бита) соответственно младшей и старшей частей двоичного кода (ДК) ЦЧ.

Входные данные: операнды A, $B \in \mathbf{Z}_{2p}$ выполняемой мультипликативной операции, представленные в МСС с базисом $\mathbf{M} = \{m_1, m_2, \ldots, m_k\}$: $A = (\alpha_1, \alpha_2, \ldots, \alpha_k)$, $B = (\beta_1, \beta_2, \ldots, \beta_k)$.

Выходные данные: аналог $€ = (€_1, €_2, \ldots, €_k)$ произведения Монтгомери $\tilde{\gamma} = |ABM_l^{-1}|_p$ $(M_l = \prod_{i=1}^{l} m_i)$ операндов A и B, удовлетворяющий условиям $€ \in \mathbf{Z}_{2p}$ и $|€|_p = \tilde{\gamma}$.

Предварительно вычисляемые данные:

- таблицы индексов и антииндексов по модулям базиса **M**, определяемые по формулам:

$$\text{TInd}i[\chi]=\begin{cases}-1 \text{ при } \chi = jm \ \ (j=\overline{0,\lfloor m\_\max/m_i\rfloor}),\\ \text{ind}_g\,|\chi|_{m_i} \text{ при } \chi\in\mathbf{Z}_{m\_\max} \text{ и } \chi\neq jm_i \ \ (j=\overline{0,\lfloor m\_\max/m_i\rfloor});\end{cases}$$

$$\text{TAInd}i[I] = |g^{|I|_{m_i-1}}|_{m_i} \ \ (I=\overline{0,2m_i-3}),$$

где $m\_\max = 2^{16}$; g – первообразный корень по модулю m_i; $i=\overline{1,k}$;

- таблицы остатков, формируемые по правилам: Tres_UP$i[S_1]$= $=s_1=|E\_0S_1|_{m_i}$ $(S_1=\overline{0,E\_1-1}; E\_0=2^{b-0};\ E\_1=2^{b-1};\ i=\overline{1,k})$; TRes$i[s]$ = $|s|_{m_i}$ $(s=\overline{0,m_i+E\_0-1};\ i=\overline{1,k})$; TRes_Norm$i[\chi]$=$\left||M_{i,l-1}^{-1}|_{m_i}\chi\right|_{m_i}$ $(M_{i.l-1}=M_{l-1}/m_i;$ $M_{l-1}=\prod_{j=1}^{l-1}m_j;\ \chi=\overline{0,m_i-1};\qquad i=\overline{1,l-1});$ _TRes_Norm$i[\chi]$=$\left|\left|\dfrac{M_lm_i}{M_{k-1}}\right|_{m_i}\chi\right|_{m_i}$ $(\chi=\overline{0,m_i-1};\ i=\overline{l+1,k-1})$; _TMpl$i[x]$=$\left|\left|M_l^{-1}\right|_{m_i}x\right|_{m_i}$ $(x=\overline{0,2m_i-2};\ i=\overline{l+1,k})$;
- таблицы интервального индекса (ИИ): TII$i[\chi]$= $R_{i,l}(\chi)=|-m_i^{-1}|M_{i,l-1}^{-1}\chi|_{m_i}|_{m_l}=$ $R_{i,l}(\chi)=|-m_i^{-1}|M_{i,l-1}^{-1}\chi|_{m_i}|_{m_l}=||-m_i^{-1}|_{m_l}\ \text{TRes\_Norm}i[\chi]|_{m_l}$ $(\chi=\overline{0,m_i-1};\ i=\overline{1,l-1})$; TII$l[\chi]$=$||M_{l-1}^{-1}|_{m_l}\chi|_{m_l}$ $(\chi=\overline{0,m_l-1})$; _TII$i[\chi]=||-m_i^{-1}|_{m_k}$ _TRes_Norm$i[\chi]|_{m_k}$ $(\chi=\overline{0,m_i-1};\ i=\overline{l+1,k-1})$; _TII$k[\chi]$=$||M_l/M_{k-1}|_{m_k}\chi|_{m_k}$ $(\chi=\overline{0,m_k-1})$;
- таблицы расширения кода МСС с базисом $\mathbf{M}_2=\{m_{l+1},\ m_{l+2},\ ...,\ m_k\}$, отвечающие ИИ, –

$$\text{\_TE}k\_j[\chi]=\begin{cases}|C'_{k,j}\chi|_{m_j}, \text{ если } \chi<m_0,\\ |C'_{k,j}(\chi-m_k)|_{m_j}, \text{ если } \chi\geq m_0,\end{cases}\quad (C'_{k,j}=|M_{k-1}/M_l|_{m_j};\ \chi=\overline{0,m_k-1};\ j=\overline{1,l}).$$

- системные константы $c_{i,j}$ = ind$C_{i,j}$ $(i=\overline{1,l-1};\ j=\overline{l+1,k})$, $c_{l,j}$ = ind$C_{l,j}$ $(j=\overline{l+1,k})$, $c'_{i,j}$ =ind$C'_{i,j}$ $(i=\overline{l+1,k-1};\ j=\overline{1,l})$ соответственно вычетов $C_{i,j}=|M_{i,l-1}|_{m_j}$, $C_{l,j}=|M_{l-1}|_{m_j}$, $C'_{i,j}=|M_{k-1}/(M_lm_i)|_{m_j}$ по модулям m_j;
- кодовое слово (ind φ_1, ind φ_2, …, ind φ_l, ind π_{l+1}, ind π_{l+2}, …, ind π_k), определяемое согласно формулам $\text{ind}\,\varphi_i=\text{ind}\,|-1/p|_{m_i}=\text{ind}\,|(m_i-\pi_i)^{-1}|_{m_i}=$ $=\begin{cases}0 \text{ при } m_i-\pi_i=1,\\ m_i-1-\text{TInd}i[m_i-\pi_i] \text{ при } m_i-\pi_i\neq 1,\end{cases}$ $i=\overline{1,\ l}$; ind π_j = TInd$j[\pi_j]$ $(j=\overline{l+1,k})$ и записываемое в массив MIC_Ip_p (MIC_Ip_p[i]=ind φ_i, MIC_Ip_p[j]=ind π_j).

Тело МИМА-алгоритма умножения Монтгомери на основе минимального КТ

УММ_1. В МСС с базисом $\mathbf{M}_1=\{m_1,\ m_2,\ ...,\ m_l\}$ по кодам $(\alpha_1,\alpha_2,...,\alpha_l)$, $(\beta_1,\beta_2,...,\beta_l)$ и модулярно-индексному коду (ind φ_1, ind φ_2, …, ind φ_l) величины F = $=|-p^{-1}|_{M_l}=(\varphi_1,\varphi_2,...,\varphi_l)$ получить код $(\delta_1,\delta_2,...,\delta_l)$ ЦЧ $D=|ABF|_{M_l}$, находя для всех $i=\overline{1,l}$ $\quad s_j=\text{TInd}i[\alpha_i]\text{TInd}i[\beta_i]+\text{MIC\_Ip\_p}[i]=\quad \text{ind}\alpha_i+\text{ind}\beta_i+\text{ind}\varphi_i$, $\quad s'_i=s_i-m_i+1$ и применяя формулу

$$\delta_i = \begin{cases} \mathrm{TAInd}i[s_i], & \text{если } s'_i < 0, \\ \mathrm{TAInd}i[s'_i], & \text{если } s'_i \geq 0. \end{cases}$$

УММ_2. Определить компьютерный ИИ $\tilde{I}_l(\tilde{D}) = \tilde{I}_l(D)$ ЦЧ $\tilde{D}$, выполняя операционную последовательность:

$$\Big\langle s_l = \sum_{i=1}^{l} \mathrm{TII}i[\delta_i];\ s_l^{(0)} = s_l \wedge \mathrm{Mask\_0},\ s_l^{(1)} = s_l >> b\_0;$$

$$\tilde{I}_l(\tilde{D}) = \eta_l = \mathrm{TRes}l[s_l^{(0)} + \mathrm{TRes\_UP}l[s_l^{(1)}]]\Big\rangle\ (Mask\_0 = 2^{b-0} - 1).$$

УММ_3. Рассчитать цифры МК ($\tilde{\delta}_{l+1}, \tilde{\delta}_{l+2}, ..., \tilde{\delta}_k$) ЦЧ $\tilde{D}$ по базису $\mathbf{M}_2 = \{m_{l+1}, m_{l+2}, ..., m_k\}$, реализуя операцию ЕС($\tilde{D}; \mathbf{M}_1, \mathbf{M}_2$) по схеме:

$$\Big\langle s_j = \sum_{i=1}^{l-1} \mathrm{TAInd}j[c_{i,j} + \mathrm{TInd}j[\mathrm{TRes\_Norm}i[\delta_i]]] + \mathrm{TAind}j[c_{l,j} + \mathrm{TInd}j[\eta_l]];$$

$$s_j^{(0)} = s_j \wedge \mathrm{Mask\_0},\ s_j^{(1)} = s_j >> b\_0;\ \tilde{\delta}_j = \mathrm{TRes}j[s_j^{(0)} + \mathrm{TRes\_UP}j[s_j^{(1)}]]\Big\rangle$$

$$(j = \overline{l+1, k}).$$

УММ_4. В МИМСС с базисом $\mathbf{M}_2$ сформировать код ($\tilde{\gamma}_{l+1}, \tilde{\gamma}_{l+2}, ..., \tilde{\gamma}_k$) ЦЧ $\tilde{\gamma} = \tilde{C} / M_l = = M_l^{-1}(AB + \tilde{D}p)$ по правилу:

$$\tilde{\gamma}_j = \_\mathrm{TMpl}j[\mathrm{TA} \mid \mathrm{Ind}j[\mathrm{TInd}j[\alpha_j] + \mathrm{TInd}j[\beta_j]] +$$

$$+ \mathrm{TAInd}j[\mathrm{TInd}j[\tilde{\delta}_j] + \mathrm{MIC\_Ip\_p}[j]]]] = |M_l^{-1}(|\alpha_j\beta_j|_{m_j} + |\tilde{\delta}_j\pi_j|_{m_j})|_{m_j}\ (\alpha_j, \beta_j, \tilde{\delta}_j \neq 0;\ j = \overline{l+1, k}).$$

$$\tilde{\gamma}_j = \_\mathrm{TMpl}j[\mathrm{TA} \mid \mathrm{Ind}j[\mathrm{TInd}j[\alpha_j] + \mathrm{TInd}j[\beta_j]] + \mathrm{TAInd}j[\mathrm{TInd}j[\tilde{\delta}_j] + \mathrm{MIC\_Ip\_p}[j]]]] =$$

$= |M_l^{-1}(|\alpha_j\beta_j|_{m_j} + |\tilde{\delta}_j\pi_j|_{m_j})|_{m_j}$ $(\alpha_j, \beta_j, \tilde{\delta}_j \neq 0;\ j = \overline{l+1, k})$. При $\alpha_j = 0$ или $\beta_j = 0$ произведение $|\alpha_j\beta_j|_{m_j} = 0$, а при $\tilde{\delta} = 0$ произведение $|\tilde{\delta}_j\pi_j|_{m_j} = 0$. В этих случаях таблицы TAInd$j$ не используются.

УММ_5. Вычислить компьютерный ИИ $\tilde{I}'_{k-l}(\tilde{\gamma})$ числа $\tilde{\gamma} = (\tilde{\gamma}_{l+1}, \tilde{\gamma}_{l+2}, ..., \tilde{\gamma}_k)$ относительно базиса $\mathbf{M}_2$, выполняя операционную последовательность:

$$\Big\langle s_k = \sum_{i=l+1}^{k} \_\mathrm{TII}i[\tilde{\gamma}_i];\ s_k^{(0)} = s_k \wedge \mathrm{Mask\_0},$$

$$s_k^{(1)} = s_k >> b\_0;\ \tilde{I}'_{k-l}(\tilde{\gamma}) = \eta_k = \mathrm{TRes}k[s_k^{(0)} + \mathrm{TRes\_UP}k[s_k^{(1)}]]\Big\rangle.$$

УММ_6. Расширить минимально избыточный МК ($\tilde{\gamma}_{l+1}, \tilde{\gamma}_{l+2}, ..., \tilde{\gamma}_k$) на модули базиса $\mathbf{M}_1$ согласно схеме:

$$\Big\langle s_j = \sum_{i=l+1}^{k-1} \mathrm{TAInd}j[c'_{i,j} + \mathrm{TInd}j[\_\mathrm{TRes\_Norm}i[\tilde{\gamma}_i]]] + \_\mathrm{TE}k\_j[\eta_k];$$

$$s_j^{(0)} = s_j \wedge \mathrm{Mask\_0},\ s_j^{(1)} = s_j >> b\_0;\ \tilde{\gamma}_j = \mathrm{TRes}j[s_j^{(0)} + \mathrm{TRes\_UP}j[s_j^{(1)}]]\Big\rangle$$

$$(j = \overline{1,\ l}).$$

При $\tilde{\gamma}_i = 0$ соответствующие слагаемые в суммах s_j обращаются в 0. В этих случаях таблицы TAInd$j$ не используются.

УММ_7. Число $\tilde{\gamma} = (\tilde{\gamma}_1, \tilde{\gamma}_2, ..., \tilde{\gamma}_k)$ зафиксировать как искомый аналог нормированного произведения $\tilde{\gamma} = |ABM_l^{-1}|_p$ операндов A и B по модулю p и завершить работу алгоритма.

Временные затраты на выполнения в k-процессорной системе модулярной обработки информации (СМОИ) алгоритма УММ_1−УММ_7 при использовании в каждом модульном тракте только одного сумматора ЦЧ составляют $t_{\text{УММ\_СМОИ}} = 2(k+2)t_{\text{сл}} + (k+15)t_{\text{ч}} + 2t_{\text{сд},b\_0}$, где $t_{\text{сл}}$, и $t_{\text{ч}}$ − длительности операций сложения и извлечения элемента таблицы. Реализация алгоритма УММ_1−УММ_7 на одиночной ПЭВМ занимает время $t_{\text{УММ\_ПЭВМ}} = (4l\_l + 3l + 4\_l)t_{\text{сл}} + (2l\_l + 7l + 9\_l + 8)t_{\text{ч}} + (k+2)t_{\text{сд},b\_0}$. При $t_{\text{сл}} = 2$нс, $t_{\text{ч}} = 1{,}14$ нс приведенные оценки в случае модулей p, разрядностью 1024 и 2462 бита соответственно дают $t_{\text{УММ\_СМОИ}} = 693{,}3$ нс, $t_{\text{УММ\_ПЭВМ}} = 45537{,}72$ нс и $t_{\text{УММ\_СМОИ}} = 1618{,}5$ нс, $t_{\text{УММ\_ПЭВМ}} = 251983{,}32$ нс.

Список литературы

1. Kawamura, S. Cox-Rower architecture for fast parallel Montgomery multiplication / Shin-ichi Kawamura, Masanobu Koike, Fumihiko Sano, Atsushi Shimbo // Eurocrypt 2000, LNCS. – Vol. 1807. – Berlin, 2000. – P. 523 – 538.
2. Bajard, J.-C. A Full RNS Implementation of RSA / J.-C. Bajard, L. Imbert // IEEE Trans. Comp. – 2004. – Vol. 53, № 6. – P. 769 – 774.
3. Lim, Z. An RNS-Enhanced microprocessor implementation of public key cryptography / Z. Lim, B.J. Phillips // Signals, Systems and Computers.-2007.- ACSSC 2007. Conf. Rec. of the forte-first Asilomar Conf. – 4–7 nov., 2007. – P. 1430 – 1434.
4. Коляда, А.А. Умножение по большим модулям с использованием минимально избыточной модулярной схемы Монтгомери / А.А. Коляда, А.Ф. Чернявский // Информатика. – 2010. – № 3. – С. 31 − 48.
5. Чернявский, А.Ф. Умножение по большим модулям методом Монтгомери с применением минимально избыточной модулярной арифметики [Текст] / А.Ф. Чернявский, А.А. Коляда, Н.А. Коляда и др. // Нейрокомпьютеры: разраб., применение. – 2010. − № 9. – Москва, 2010. − С. 3 – 8.
6. Каленик, А.Н. Умножение и возведение в степень по большим модулям с использованием минимально избыточной модулярной арифметики / А.Н. Каленик, А.А. Коляда, Н.А. Коляда, А.Ф. Чернявский, Е.В. Шабинская // Информационные технологии. – 2012. – № 4. – С. 37 – 44.

The new algorithm of multiplication on the big module, based on the optimized minimally redundant modular Montgomery's scheme and having table-summarized configuration is presented. High speed is provided when using the minimum set of tables.

Каленик А.Н., соискатель кафедры интеллектуальных систем БГУ, Минск, Беларусь, E-mail: andrei.kalenik@gmail.com.

Коляда А.А., г.н.с. НИИПФП им. А.Н.Севченко БГУ, д.ф.-м.н., Минск, Беларусь, e-mail: razan@tut.by.

Мазуренко П.А., аспирант кафедры интеллектуальных систем БГУ, Минск, Беларусь, E-mail: mazurenkopa@gmail.com.

Шабинская Е.В., с.н.с. НИИПФП им. А.Н.Севченко БГУ, к.т.н., Минск, Беларусь, e-mail: shabinskaya@rambler.ru.