УДК 681.04

В.А. Краснобаев, С.А. Кошман

Харьковский национальный технический университет сельского хозяйства имени Петра Василенко, Харьков

МЕТОД БЫСТРОЙ РЕАЛИЗАЦИИ КРИПТОГРАФИЧЕСКИХ ПРЕОБРАЗОВАНИЙ НА ОСНОВЕ ПОРАЗРЯДНОЙ ТАБЛИЧНОЙ РЕАЛИЗАЦИИ

В статье предложен метод повышения быстродействия реализации криптографических преобразований, основанный на использовании основных свойств непозиционной системы счисления в остаточных классах.

Ключевые слова: *система счисления, система остаточных классов, табличная арифметика, специальный код табличного представления операндов.*

Введение

Постановка проблемы. Современные криптопреобразования с открытым ключом основываются на преобразованиях, на алгебраических кривых (эллиптические кривые (ЭК), гиперэллиптические кривые (ГЭК), кривые Пикарда (КП) и суперэллиптические кривые (СУ)). Тенденция развития криптографических методов направлена на увеличения длины ключей, что в свою очередь приводит к снижению быстродействия криптографических преобразований с открытым ключом. Это особенно критично для обеспечения заданного уровня стойкости при реализации криптопреобразований на ЭК в специальных системах и устройствах, где есть существенные ограничения по объему памяти и массогабаритным характеристикам, т.е. в тех случаях, где нет возможности использовать мощные стационарные высокопроизводительные вычислители с большой разрядной сеткой. Данное обстоятельство обуславливает важность и актуальность поисков методов повышения производительности, надежности и достоверности криптопреобразований.

Анализ последних исследований и публикаций. Анализ методов повышения производительности СУ в якобиане ГЭК позволил теоретически обосновать и практически показать зависимость производительности реализации операций СУ в якобиане ГЭК от совокупности следующих основных характеристик: от вида реализации криптопреобразований (программная, аппаратная и программно-аппаратная); от вида алгоритма СУ дивизоров; от заданного базового поля, над которым задается данная кривая; от типа кривой; от значений коэффициентов кривой; от выбранной системы координат, в которой представлены дивизоры якобиана ГЭК (аффинная, проективная, взвешенная и смешанная); от принятого метода арифметических преобразований в якобиане и пр.

Известные методы реализации алгоритмов СУ (метод сложения дивизоров Кантора, метод Коблица, методы арифметических преобразований дивизоров в якобиане ГЭК второго, третьего и четвертого рода, методы сложения дивизоров различного веса, метод Карацубы для умножения и приведения по модулю в поле полиноминальных функций, метод, основанный на некоторых результатах "Китайской теоремы об остатках" и пр.) не всегда удовлетворяют требованиям по производительности криптопреобразований.

В тоже время, в литературе [1 – 3] показана высокая эффективность применения системы счисления в остаточных классах (СОК), при решении отдельных задач обработки цифровой информации (решение задач фильтрации, задач реализации БПФ, ДПФ и пр.) с точки зрения высокой производительности их реализации. Так, известно, что преобразование Фурье связано с вычисление полинома вида $P(x) = \sum_{i=1}^{n-1} \alpha_i x^i$.

Одно из приложений преобразования Фурье – вычисление свертки $\sum_{i=1}^{n} \alpha_i \beta_i$ двух n-мерных векторов $A = (\alpha_1, \alpha_2, ..., \alpha_n)$ и $B = (\beta_1, \beta_2, ..., \beta_n)$. В данном случае операция свертки является полным аналогом реализации арифметических операций умножения двух чисел А и В в СОК с последующим сложением компонент типа $\alpha_i \beta_i (\mathrm{mod}\ m_i) + \alpha_j \beta_j (\mathrm{mod}\ m_j)$.

В этом аспекте данное обстоятельство обуславливает важность и актуальность поисков методов повышения производительности, надежности и достоверности криптопреобразований на основе использования свойств непозиционных кодовых структур СОК.

Целью статьи является разработка метода повышения производительности выполнения криптографических преобразований на основании использования СОК, используя табличный принцип реализации арифметических операций.

Изложение основного материала

Существующие методы повышения быстродействия выполнения криптопреобразований оперируют с данными, представленными в ПСС, где выполнение арифметической операции предполагает последовательную обработку разрядов операндов по правилам, определяемым содержанием данной операции, и не может быть закончено до тех пор, пока не будут последовательно определены значения всех промежуточных результатов с учетом всех связей между разрядами. Таким образом, ПСС, в которых представляется и обрабатывается информация в современных вычислительных машинах (ВМ), обладают существенным недостатком – наличием межразрядных связей, которые накладывают свой отпечаток на методы реализации арифметических операций, усложняют аппаратуру и ограничивают быстродействие. К тому же наличие межразрядных связей не позволяет распараллелить решаемые алгоритмы на уровне элементарных операций. Алгоритмическая связь в ПСС всех двоичных разрядов операнда между собой обусловливает тот факт, что единичный отказ или сбой схемы обработки одного двоичного разряда операционного устройства способен вызвать не однократную, а многократные ошибки в машинном слове.

Поэтому естественно изыскание возможностей построения такой арифметики, в которой бы поразрядные связи отсутствовали. В этом плане обращает на себя внимание система счисления в остаточных классах.

Об эффективность использования СОК можно судить, анализируя основные ее свойства.

1. Независимость остатков. Это свойство дает возможность построить ВМ в виде набора информационно независимых вычислительных трактов, т.е. отдельных вычислительных устройств (ВУ), функционирующих по своему определенному модулю m_i в СОК. При этом ошибки, возникающие за счет отказов (сбоев) схем двоичных разрядов в произвольном вычислительном тракте ВУ, не «размножаются» в соседние тракты, а остаются в пределах одного остатка, что дает возможность повысить достоверность вычислений в СОК.

Свойство независимости остатков порождает возможность организации параллельной обработки информации представленной в остатках.

2. Равноправность остатков. Любой остаток a_i числа A_k в СОК несет информацию обо всем исходном числе, что дает возможность программными методами заменить искаженный тракт по модулю m_i на исправный (контрольный) тракт по модулю m_j ($m_i < m_j$), не прерывая решения задачи.

3. Малоразрядность остатков. Это свойство позволяет существенно повысить быстродействие выполнения арифметических операций как за счет малоразрядности вычислительных трактов ВУ, так и за счет возможности применения (в отличие от ПСС) табличной арифметики, где арифметические операции сложения, вычитания и умножения выполняются практически в один такт.

В общем случае количество логических схем совпадения (элементов И) в узлах ПЗУ при табличном методе реализации арифметических операций определяется как $N_{СОК} = \sum_{i=1}^{n} m_i^2$. В то же время, учитывая избыточность таблиц при реализации ВУ в СОК, представляется рациональным использовать методы, которые позволяют сократить количество оборудования таблиц ПЗУ.

Так в литературе [3 – 5] показано, что таблица реализации арифметической операции модульного умножения симметрична относительно вертикали, горизонтали и диагоналей.

В связи с этим рассмотрим метод, который позволяет восстановить числовые данные таблицы модульного умножения $a_i\beta_i (\mathrm{mod}\, m_i)$, посредством использования кода табличного умножения (КТУ), имея числовую информацию о ее четвертой части. Так если заданы два операнда в КТУ $a_i = (\gamma_a, a_i')$, $\beta_i = (\gamma_\beta, \beta_i')$, то для, того чтобы получить произведение этих чисел по модулю m_i, достаточно получить произведение $a_i'\beta_i' (\mathrm{mod}\, m_i)$ и инвертировать его обобщенный индекс γ_i в случае, если γ_a отлично от γ_β, т.е.

$$a_i\beta_i (\mathrm{mod}\, m_i) = (\gamma_i, a_i'\beta_i (\mathrm{mod}\, m_i)),$$

где $\gamma_i = \begin{cases} \overline{\gamma_i}, \text{ если } \gamma_a \neq \gamma_\beta, \\ \gamma, \text{ если } \gamma_a = \gamma_\beta. \end{cases}$ – обобщенный индекс КТУ.

До настоящего времени вопросы эффективной реализации, посредством КТУ, арифметических операций сложения и вычитания не освещены. Основная трудность заключается в том, что довольно сложно синтезировать алгоритмы модульных операций в связи с тем, что таблицы выполнения модульных операций различны по своей цифровой структуре.

Однако совершенно иные результаты можно получить, введя понятие специального кода табличного представления операндов (СКТПО) и исследуя возможности реализации одной модульной операции посредством таблицы, реализующих ей обратную.

При исследовании цифровых свойств таблиц модульных операций сложения и вычитания доказано соотношение

$$\left\lfloor (\gamma_a, a_i') + (\gamma_\beta, \beta_i') \right\rfloor + \\ + \left\{ \left[m_i - (\gamma_a, a_i') \right] - (\gamma_\beta, \beta_i') \right\} = 0 (\mathrm{mod}\, m_i), \quad (1)$$

где $a_i = (\gamma_a, a_i')$, $\beta_i = (\gamma_b, \beta_i')$ – входные операнды, представленные в СКТПО.

Из выражения (1) можно сделать вывод, что для получения результата операции модульного сложения, посредством СКТПО, достаточно знать результат модульного вычитания т.е.

$$(\gamma_a, a_i') + (\gamma_b, b_i') = m_i - \left\{ \left[m_i - (\gamma_a, a_i') \right] - (\gamma_b, b_i') \right\}, \quad (2)$$

а для определения результата операции модульного вычитания достаточно знать результат модульного сложения т.е.

$$(\gamma_a, a_i') - (\gamma_\beta, \beta_i') = \left\{ (\gamma_a, a_i') + \left[m_i (\gamma_\beta, \beta_i') \right] \right\}, \quad (3)$$

Таким образом возникает возможность эффективно (с точки зрения уменьшения оборудования ПЗУ) использовать специальный кодатабличного представления операндов для реализации модульных операций сложения и вычитания.

Ниже на рис. 1 и рис. 2. представлены методы, разработанные на основании выражений 2 и 3.

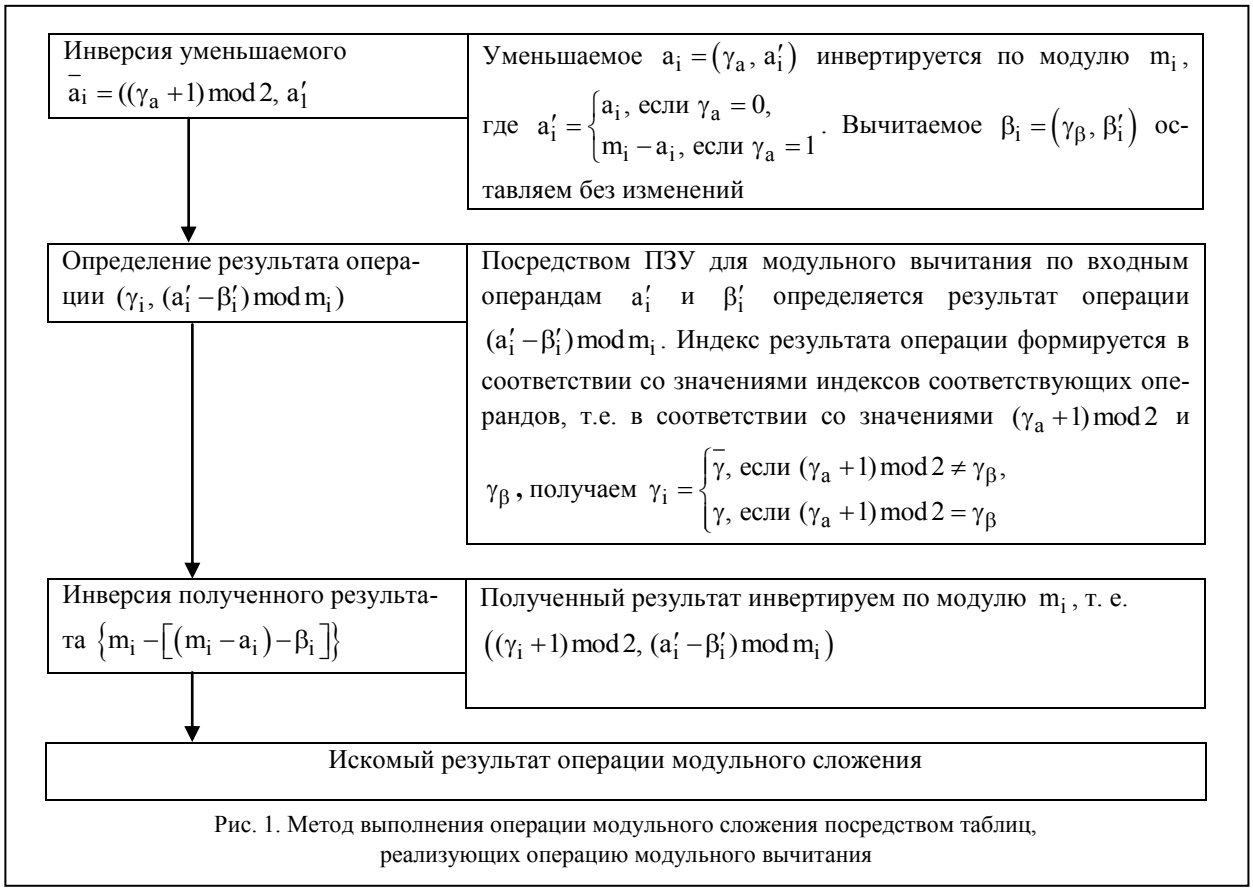

Рис. 1. Метод выполнения операции модульного сложения посредством таблиц, реализующих операцию модульного вычитания

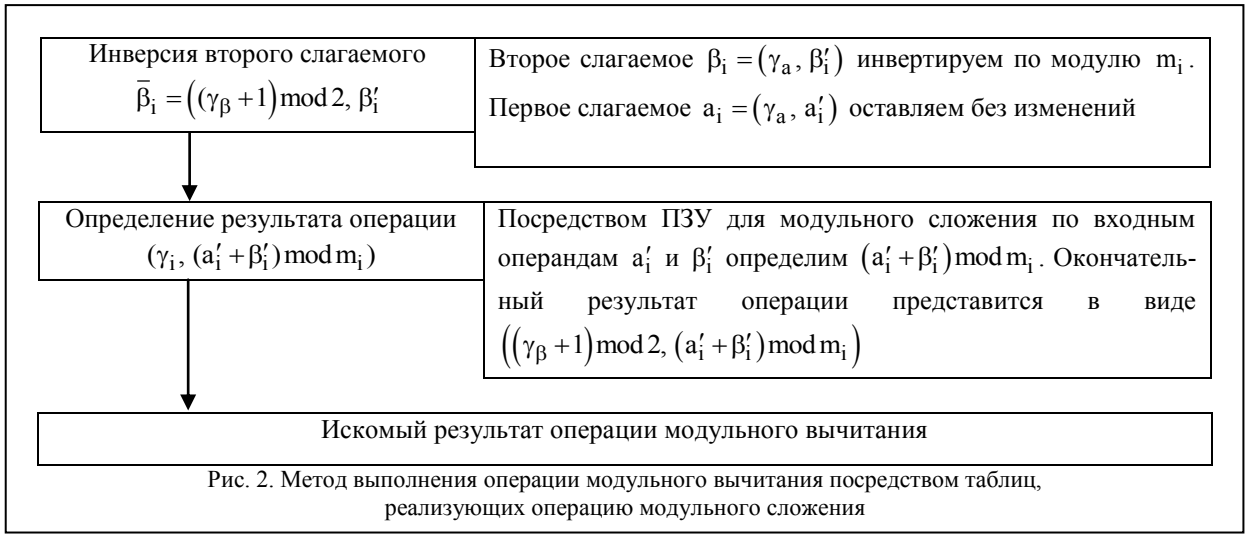

Рис. 2. Метод выполнения операции модульного вычитания посредством таблиц, реализующих операцию модульного сложения

Анализируя представленные методы можно сделать вывод, что при совместной реализации арифметических операций сложение и вычитание, метод представленный на рис. 2 позволяет за меньшее время и с меньшими аппаратурными затратами, по сравнению с методом представленным на рис. 1 реализовать заданную в СОК арифметическую операцию вычитания.

Для построения таблиц основных арифметических операций представляется наиболее эффектным применение метода специального кодирования, который описан выше, и позволяет одновременно уменьшить размер таблиц сложения, вычитания и умножения в четыре раза [6 – 8].

При реализации операций табличными методами в ряде случаев возможно дополнительное уменьшение оборудования за счет того, что строится не единая таблица, реализующая результат в двоичном коде, а k более мелких таблиц, реализующих ответы по каждому из k разрядов результата, где k – разрядность регистра, необходимая для хранения цифры по рассматриваемому основанию $k = \left[\log_2(m_i - 1)\right] + 1$. При этом довольно часто имеет место унификация таблиц, т. е. сокращение количества различных типов таблиц, необходимых для реализации арифметического устройства.

В табл. 1 представлена схема реализации обобщенной $\otimes$ арифметической операции (где $\otimes$ – умножение, сложение и вычитание) по произвольному модулю m_i, а так же указана симметрия относительно вертикали, горизонтали и диагоналей.

С учетом симметрии таблиц реализации основных арифметических операций (умножения, сложения и вычитания), а также на основе рассмотренных выше методов сокращения таблиц, в табл.. 2 сводятся данные ¼ части табл. 1, и в частности, ее второй квадрант, которые для удобства счета могут быть представлены десятичным кодом.

В табл. 3 в двоичном коде представлены числовые данные второго квадранта табл. 1.

На основании табл. 3 сведем в табл. 4 значения соответствующие первому (младшему) разряду результата. Причем для реализации ВУ будем использовать только те ячейки (узлы) табл. 4, которые соответствуют единичным значениям младшего разряда результата.

Аналогичным образом, основываясь на табл. 3, сведем в табл. 5 и 6 значения соответствующие второму и k (старшему) разрядам результата обобщенной арифметической операции.

Несмотря на то, что уменьшен размер каждой таблицы и увеличено их количество, в целом имеет место выигрыш в количестве оборудования, поскольку до предела сокращена избыточность таблиц и, как видим, реализуются только узлы таблицы, которые соответствуют значащим разрядам результата. Так как результат операции представляется машинным кодом, то нет необходимости в логических элементах, формирующих индекс СКТПО.

Таблица 1

Таблица реализации обобщенной арифметической операции модулю m

a / b	a_0	…	$a_{\frac{m-1}{2}}$	$a_{\frac{m+1}{2}}$	…	a_{m-1}
b_0	$(a_0 \otimes b_0) \bmod m = c_{00}$	…	$(a_{\frac{m-1}{2}} \otimes b_0) \bmod m = c_{(\frac{m-1}{2})0}$	$(a_{\frac{m+1}{2}} \otimes b_0) \bmod m = c_{(\frac{m+1}{2})0}$	…	$(a_{m-1} \otimes b_0) \bmod m = c_{(m-1)0}$
…	…	…	…	…	…	…
$b_{\frac{m-1}{2}}$	$(a_0 \otimes b_{\frac{m-1}{2}}) \bmod m = c_{0(\frac{m-1}{2})}$	…	$(a_{\frac{m-1}{2}} \otimes b_{\frac{m-1}{2}}) \bmod m = c_{(\frac{m-1}{2})(\frac{m-1}{2})}$	$(a_{\frac{m+1}{2}} \otimes b_{\frac{m-1}{2}}) \bmod m = c_{(\frac{m+1}{2})(\frac{m-1}{2})}$	…	$(a_{m-1} \otimes b_{\frac{m-1}{2}}) \bmod m = c_{(m-1)(\frac{m-1}{2})}$
$b_{\frac{m+1}{2}}$	$(a_0 \otimes b_{\frac{m+1}{2}}) \bmod m = c_{0(\frac{m+1}{2})}$	…	$(a_{\frac{m-1}{2}} \otimes b_{\frac{m+1}{2}}) \bmod m = c_{(\frac{m-1}{2})(\frac{m+1}{2})}$	$(a_{\frac{m+1}{2}} \otimes b_{\frac{m+1}{2}}) \bmod m = c_{(\frac{m+1}{2})(\frac{m+1}{2})}$	…	$(a_{m-1} \otimes b_{\frac{m+1}{2}}) \bmod m = c_{(m-1)(\frac{m+1}{2})}$
…	…	…	…	…	…	…
b_{m-1}	$(a_0 \otimes b_{m-1}) \bmod m = c_{0(m-1)}$	…	$(a_{\frac{m-1}{2}} \otimes b_{m-1}) \bmod m = c_{(\frac{m-1}{2})(m-1)}$	$(a_{\frac{m+1}{2}} \otimes b_{m-1}) \bmod m = c_{(\frac{m+1}{2})(m-1)}$	…	$(a_{m-1} \otimes b_{m-1}) \bmod m = c_{(m-1)(m-1)}$

Таблица 2

Второй квадрант таблицы 1

b \ a		a_0	a_1	…	$a_{\frac{m-1}{2}}$
			a_{m-1}	…	$a_{\frac{m+1}{2}}$
b_0		$(a_0 \otimes b_0) \bmod m = c_{00}$	$(a_1 \otimes b_0) \bmod m = c_{10}$		$(a_{\frac{m-1}{2}} \otimes b_0) \bmod m = c_{(\frac{m-1}{2})0}$
b_1	b_{m-1}	$(a_0 \otimes b_1) \bmod m = c_{01}$	$(a_1 \otimes b_1) \bmod m = c_{11}$	…	$(a_{\frac{m-1}{2}} \otimes b_1) \bmod m = c_{(\frac{m-1}{2})1}$
b_2	b_{m-2}	$(a_0 \otimes b_2) \bmod m = c_{02}$	$(a_1 \otimes b_2) \bmod m = c_{12}$	…	$(a_{\frac{m-1}{2}} \otimes b_2) \bmod m = c_{(\frac{m-1}{2})2}$
…	…	…	…	…	…
$b_{\frac{m-1}{2}}$	$b_{\frac{m+1}{2}}$	$(a_0 \otimes b_{\frac{m-1}{2}}) \bmod m = c_{0(\frac{m-1}{2})}$	$(a_1 \otimes b_{\frac{m-1}{2}}) \bmod m = c_{1(\frac{m-1}{2}}$	…	$(a_{\frac{m-1}{2}} \otimes b_{\frac{m-1}{2}}) \bmod m = c_{(\frac{m-1}{2})(\frac{m-1}{2}}$

Таблица 3

Данные второго квадранта таблицы 1 представленные в двоичном коде

b \ a		a_0	a_1	…	$a_{\frac{m-1}{2}}$
			a_{m-1}	…	$a_{\frac{m+1}{2}}$
b_0		$c_{00_k}, c_{00_{(k-1)}}, \cdots$ $\ldots, c_{00_1}, c_{00_0}$	$c_{10_k}, c_{10_{(k-1)}}, \cdots$ $\ldots, c_{10_1}, c_{10_0}$	…	$c_{(\frac{m-1}{2})0_k}, c_{(\frac{m-1}{2})0_{(k-1)}}, \cdots$ $\ldots, c_{(\frac{m-1}{2})0_1}, c_{(\frac{m-1}{2})0_0}$
b_1	b_{m-1}	$c_{01_k}, c_{01_{(k-1)}}, \cdots$ $\ldots, c_{01_1}, c_{01_0}$	$c_{11_k}, c_{11_{(k-1)}}, \cdots$ $\ldots, c_{11_1}, c_{11_0}$	…	$c_{(\frac{m-1}{2})1_k}, c_{(\frac{m-1}{2})1_{(k-1)}}, \cdots$ $\ldots, c_{(\frac{m-1}{2})1_1}, c_{(\frac{m-1}{2})1_0}$
b_2	b_{m-2}	$c_{02_k}, c_{02_{(k-1)}}, \cdots$ $\ldots, c_{02_1}, c_{02_0}$	$c_{12_k}, c_{12_{(k-1)}}, \cdots$ $\ldots, c_{12_1}, c_{12_0}$	…	$c_{(\frac{m-1}{2})2_k}, c_{(\frac{m-1}{2})2(k-1)}, \cdots$ $\ldots, c_{(\frac{m-1}{2})2_1}, c_{(\frac{m-1}{2})2_0}$
…	…	…	…	…	…
$b_{\frac{m-1}{2}}$	$b_{\frac{m+1}{2}}$	$c_{0(\frac{m-1}{2})_k}, c_{0(\frac{m-1}{2})_{(k-1)}}, \cdots$ $\ldots, c_{0(\frac{m-1}{2})_1}, c_{0(\frac{m-1}{2})_0}$	$c_{1(\frac{m-1}{2})_k}, c_{1(\frac{m-1}{2})_{(k-1)}}, \cdots$ $\ldots, c_{1(\frac{m-1}{2})_1}, c_{1(\frac{m-1}{2})_0}$	…	$c_{(\frac{m-1}{2})(\frac{m-1}{2})_k}, c_{(\frac{m-1}{2})(\frac{m-1}{2})_{(k-1)}}, \cdots$ $\ldots, c_{(\frac{m-1}{2})(\frac{m-1}{2})_1}, c_{(\frac{m-1}{2})(\frac{m-1}{2})_0}$

Таблица 4

Значения первого разряда результата таблицы 3

b \ a		a_0	a_1	…	$a_{\frac{m-1}{2}}$
			a_{m-1}	…	$a_{\frac{m+1}{2}}$
b_0		c_{00_0}	c_{10_0}	…	$c_{(\frac{m-1}{2})0_0}$
b_1	b_{m-1}	c_{01_0}	c_{11_0}	…	$c_{(\frac{m-1}{2})1_0}$
b_2	b_{m-2}	c_{02_0}	c_{12_0}	…	$c_{(\frac{m-1}{2})2_0}$
…	…	…	…	…	…
$b_{\frac{m-1}{2}}$	$b_{\frac{m+1}{2}}$	$c_{0(\frac{m-1}{2})_0}$	$c_{1(\frac{m-1}{2})_0}$	…	$c_{(\frac{m-1}{2})(\frac{m-1}{2})_0}$

Выводы

В данной статье предложен метод существенного повышения быстродействия реализации крип-

тографических преобразований в полях Галуа. Данный метод основан на использовании непозиционной системы счисления в остаточных классах.

Таблица 5

Значения второго разряда результата таблицы 3

b \ a		a_0	a_1	…	$a_{\frac{m-1}{2}}$
			a_{m-1}	…	$a_{\frac{m+1}{2}}$
b_0		c_{00_1}	c_{10_1}	…	$c_{(\frac{m-1}{2})0_1}$
b_1	b_{m-1}	c_{01_1}	c_{11_1}	…	$c_{(\frac{m-1}{2})1_1}$
b_2	b_{m-2}	c_{02_1}	c_{12_1}	…	$c_{(\frac{m-1}{2})2_1}$
…	…	…	…	…	…
$b_{\frac{m-1}{2}}$	$b_{\frac{m+1}{2}}$	$c_{0(\frac{m-1}{2})_1}$	$c_{1(\frac{m-1}{2})_1}$	…	$c_{(\frac{m-1}{2})(\frac{m-1}{2})_1}$

Таблица 6

Значения k разряда результата таблицы 3

b \ a		a_0	a_1	…	$a_{\frac{m-1}{2}}$
			a_{m-1}	…	$a_{\frac{m+1}{2}}$
b_0		c_{00_k}	c_{10_k}	…	$c_{(\frac{m-1}{2})0_k}$
b_1	b_{m-1}	c_{01_k}	c_{11_k}	…	$c_{(\frac{m-1}{2})1_k}$
b_2	b_{m-2}	c_{02_k}	c_{12_k}	…	$c_{(\frac{m-1}{2})2_k}$
…	…	…	…	…	…
$b_{\frac{m-1}{2}}$	$b_{\frac{m+1}{2}}$	$c_{0(\frac{m-1}{2})_k}$	$c_{1(\frac{m-1}{2})_k}$	…	$c_{(\frac{m-1}{2})(\frac{m-1}{2})_k}$

Использование основных свойств СОК и позволяет организовать процесс реализации модульных операций в криптографических задачах. Производительность вычислений повышается за счет использования табличного принципа реализации арифметических операций в СОК, а также путем введения и использования специального кода табличного представления операндов с учетом симметрии таблиц.

Список литературы

1. Шнайер Б. Прикладная криптографія / Б. Шнайер. – М.: Триумф, 2002. – 797 с.

2. Горбенко И.Д. Криптоанализ криптографических преобразований в группах точек эллиптических кривых методом полларда / И.Д. Горбенко, С.И. Збитнев, А.А. Полеков // Радиотехника: Всеукр. межвед. научн-тех. сб. – 2001. – Вып. 119. – С. 43-50.

3. Krasnobayev V.A. Method for Realization of Transformations in Public-Key Cryptography / V.A. Krasnobayev // Telecommunications and Radio Engineering (USA). – 2007. – Vol. 66, Issue 17. – P. 1559-1572.

4. Акушский И.Я. Машинная арифметика в остаточных классах / И.Я. Акушский, Д.И. Юдицкий. – М.: Сов. радио, 1968. – 440 с.

5. Кошман С. А. Табличный метод обработки цифровой информации в классе вычетов / С.А. Кошман, С.Н. Деренько, В.А. Краснобаев // Радіоелектронні і комп'ютерні системи. – 2006. – № 5 (17). – С. 171-175.

6. Фурман И.А. Вариант синтеза процессора в системе остаточных классов / И.А. Фурман, С.А. Кошман, В.А. Краснобаев // Радиотехника и Информатика. – 2003. – № 2. – С. 94-96.

7. Фурман І.О. Аналіз табличних алгоритмів реалізації модульних операцій у автоматизованих системах обробки цифрової інформації / І.О. Фурман, В.А. Краснобаєв, С.О. Кошман // Проблеми енергозабезпечення та енергозбереження в АПК України: Вісник ХДТУСГ. – Х., 2004. – Вип. 27, т. 2. – С. 174-178.

8. Кошман С.О. Диверсність табличних методів реалізації арифметичних операцій у системі залишкових класів / С.О. Кошман, В.І. Барсов, В.А. Краснобаєв // Проблеми енергозабезпечення та енергозбереження в АПК України: Вісник ХНТУСГ ім. Петра Василенка. – Х., 2008. – Вип. 73, т. 2. – С. 70-72.

Поступила в редколлегию 12.03.2009

Рецензент: д-р техн. наук, проф. И.А. Фурман, Харьковский национальный технический университет сельского хозяйства имени Петра Василенко, Украина.

МЕТОД ШВИДКОЇ РЕАЛІЗАЦІЇ КРИПТОГРАФІЧНИХ ПЕРЕТВОРЕНЬ НА ОСНОВІ ПОРОЗРЯДНОЇ ТАБЛИЧНОЇ РЕАЛІЗАЦІЇ

В.А. Краснобаєв, С.О. Кошман

У статті запропонований метод підвищення швидкодії реалізації криптографічних перетворень, заснований на використанні основних властивостей непозиційної системи числення в залишкових классах.

Ключові слова: *система числення, система залишкових класів, таблична арифметика, спеціальний код табличного подання операндів.*

METHOD OF RAPID REALIZATION OF CRYPTOGRAPHIC TRANSFORMATIONS ON THE BASIS OF DIGIT-BY-DIGIT TABULAR REALIZATION

V.A. Krasnobaev, S.A. Koshman

The method of increase of fast-acting of realization of cryptographic transformations is offered in the article, based on the use of basic properties of the unposition number system in remaining classes.

Keywords*: system of the numeration(reckoning), system of the remaining classes, tabular arithmetic, special code of the tabular presentation operand.*