УДК 004.056.3

АРИФМЕТИКА ДВОИЧНОГО ПОЛЯ ГАЛУА НА БАЗЕ БЫСТРОГО УМНОЖЕНИЯ И ИНВЕРТИРОВАНИЯ ЭЛЕМЕНТОВ ПОЛЯ И ЕЕ АППАРАТНАЯ РЕАЛИЗАЦИЯ

Рахман П.А.

ФГБОУ ВПО «Уфимский государственный нефтяной технический университет», филиал в г. Стерлитамаке, e-mail: pavelar@yandex.ru

В данной статье рассматриваются поля Галуа GF(2^m) характеристики 2 и их арифметика. Рассматриваются операции сложения и вычитания, операция умножения элементов на базе прямых формул для расчета коэффициентов многочлена-произведения, а также операции деления на базе умножения на мультипликативную инверсию элемента-множителя. Приводится математическое описание и аппаратная реализация схемы быстрого умножения на базе двухвходовых элементов «И» и многовходовых сумматоров по модулю 2. Также приводится схема аппаратной реализации процессора, реализующего арифметические операции в поле Галуа на базе сумматоров по модулю 2, постоянного запоминающего устройства для хранения мультипликативных инверсий элементов, мультиплексора и схемы быстрого умножения элементов.

Ключевые слова: поле Галуа, быстрое умножение, таблица инверсий, арифметический процессор

ARITHMETIC OF BINARY GALOIS FIELD BASED ON FAST MULTIPLICATION AND INVERSION OF FIELD ELEMENTS AND ITS HARDWARE IMPLEMENTATION

Rahman P.A.

Ufa State Petroleum Technological University, Sterlitamak branch, e-mail: pavelar@yandex.ru

This paper deals with Galois fields GF(2^m) with characteristic 2 and their arithmetic. The addition, and subtraction operations of elements, and multiplication operation based on direct formulas for calculation of result-polynomial coefficients, and division operation based on multiplication by inverse of multiplier-element are also observed. Mathematical background and hardware implementation for fast multiplication based on 2-input «AND» elements and multi-input modulo 2 adders are also discussed. Hardware implementation of processor for Galois field arithmetic, based on modulo 2 adders, read-only memory for inversion table, multiplexor and circuit for fast multiplication of elements, are also overviewed.

Keywords: Galois field, fast multiplication, inversion table, arithmetic processor

В мире информационных технологий конечные поля Галуа $GF(2^m)$ имеют огромное практическое значение [1]. В частности, важнейшие алгоритмы обнаружения и исправления искажения информации в системах хранения и сетях передачи данных, использующие коды Рида-Соломона, а также криптографические алгоритмы (например, AES – Advanced Encryption Standard), защищающие информацию от несанкционированного доступа, базируются на арифметике конечных полей Галуа $GF(2^m)$.

Однако для эффективной программной и аппаратной реализации алгоритмов, также необходима быстродействующая аппаратная реализация арифметики поля Галуа $GF(2^m)$. В частности, для ускорения операций умножения и деления элементов необходимы специальные подходы к разработке быстрой параллельной схемы умножения элементов и нахождения мультипликативной инверсии элемента для операции деления.

В рамках научных исследований в области надежности систем хранения, передачи и обработки данных [3–9], а также методов информационного резервирования [2, 10], автором была исследована эффективная аппаратная схема быстрого умножения, и на базе нее была разработана схема арифметического процессора для поля Галуа $GF(2^m)$.

Арифметика поля Галуа $GF(2^m)$. Поле Галуа $GF(2^m)$, по определению являющееся полем многочленов вида $a(x) = a_{m-1}x^{m-1} + \ldots + a_1x + a_0$, $a_i \in \{0,1\}$, образуется на базе простого поля Галуа $GF(2)$ и нормированного примитивного неприводимого многочлена m-й степени: $p(x) = x^m + p_{m-1}x^{m-1} + p_1x + p_0$, $p_i \in \{0,1\}$. Особо отметим, что элементы поля можно также рассматривать как m-разрядные двоичные числа $(a)_2 = (a_{m-1}\ldots a_1a_0)_2$, и более того, для компактной формы представления записывать двоичные эквиваленты элементов поля в десятичной форме $(a)_{10}$.

Например, поле Галуа $GF(2^4)$ образуется при помощи неприводимого многочлена $p(x) = x^4 + x + 1$, и его элементы можно рассматривать как многочлены, так и соответствующие двоичные и десятичные эквиваленты:

$GF(2^4)$:	$a(x)$:	0	1	x	$x+1$	x^2	x^2+1	x^2+x	x^2+x+1
	$(a)_2$:	0000	0001	0010	0011	0100	0101	0110	0111
	$(a)_{10}$:	0	1	2	3	4	5	6	7

$a(x)$:	x^3	x^3+1	x^3+x	x^3+x+1	x^3+x^2	x^3+x^2+1	x^3+x^2+x	x^3+x^2+x+1
$(a)_2$:	1000	1001	1010	1011	1100	1101	1110	1111
$(a)_{10}$:	8	9	10	11	12	13	14	15

Отметим, что в базовом простом поле $GF(2)$ для элемента 0 обратным элементом по сложению является сам элемент 0, также как и для элемента 1 обратным элементом по сложению является сам элемент 1. Соответственно, как сложение, так и вычитание элементов простого поля $GF(2)$ фактически сводятся к одной и той же операции суммирования по модулю 2, и обозначается символом $\oplus$.

Тогда, при сложении и вычитании элементов поля $GF(2^m)$ мы имеем сложение соответствующих коэффициентов многочленов по модулю 2 (при представлении в виде многочленов) или побитовое сложение по модулю 2 соответствующих разрядов двоичных чисел (при представлении в виде двоичных чисел):

$$\begin{aligned}\underbrace{a \pm b}_{GF(2^m)} &= \underbrace{(a(x) \pm b(x)) \bmod p(x)}_{GF(2)} = \underbrace{(a_{m-1} \pm b_{m-1})}_{GF(2)} x^{m-1} + \ldots + \underbrace{(a_1 \pm b_1)}_{GF(2)} x + \underbrace{(a_0 \pm b_0)}_{GF(2)} = \\ &= (a_{m-1} \oplus b_{m-1}) x^{m-1} + \ldots + (a_1 \oplus b_1) x + (a_0 \oplus b_0) = ((a_{m-1} \oplus b_{m-1}) \ldots (a_0 \oplus b_0))_2.\end{aligned} \quad (1)$$

Пример. Найдем сумму элементов расширенного поля $GF(2^4)$, представленных в виде соответствующих чисел «13» и «7» в десятичной системе счисления. Имеем,

$$\underbrace{(13)_{10} + (7)_{10}}_{GF(2^4)} = \underbrace{(1101)_2 + (0111)_2}_{GF(2^4)} = \left\{ \begin{array}{c|c|c|c} \begin{matrix}1\\ \oplus\\ 0\end{matrix} & \begin{matrix}1\\ \oplus\\ 1\end{matrix} & \begin{matrix}0\\ \oplus\\ 1\end{matrix} & \begin{matrix}1\\ \oplus\\ 1\end{matrix} \\ \hline 1 & 0 & 1 & 0 \end{array} \right\} = (1010)_2 = (10)_{10}.$$

Быстрое умножение и деление элементов поля $GF(2^m)$. Для построения быстрых и компактных умножителей следует использовать классическое определение операции умножения элементов поля Галуа $GF(2^m)$, представленных в виде многочленов с коэффициентами из простого поля $GF(2)$:

$$\forall a, b \in GF(2^m) \Rightarrow \underbrace{a \cdot b}_{GF(2^m)} = \underbrace{(a(x) \cdot b(x)) \bmod p(x)}_{GF(2)}.$$

Рассмотрим умножение элементов на примере поля Галуа $GF(2^4)$, образованного при помощи примитивного неприводимого многочлена $p(x) = x^4 + x + 1$. Имеем следующее:

$$\underbrace{a \cdot b}_{GF(2^4)} = \underbrace{((a_3x^3 + a_2x^2 + a_1x + a_0)(b_3x^3 + b_2x^2 + b_1x + b_0)) \bmod (x^4 + x + 1)}_{GF(2)}.$$

После перемножения многочленов и вычисления остатка по модулю $p(x) = x^4 + x + 1$ в общем виде получаем:

$$\underbrace{a \cdot b}_{GF(2^4)} = \underbrace{(a(x) \cdot b(x)) \bmod p(x)}_{GF(2)} = c_3x^3 + c_2x^2 + c_1x + c_0;$$

$$\begin{cases} c_3 = a_0 \cdot b_3 \oplus a_1 \cdot b_2 \oplus a_2 \cdot b_1 \oplus a_3 \cdot b_0 \oplus a_3 \cdot b_3; \\ c_2 = a_0 \cdot b_2 \oplus a_1 \cdot b_1 \oplus a_2 \cdot b_0 \oplus a_3 \cdot b_3 \oplus a_2 \cdot b_3 \oplus a_3 \cdot b_2; \\ c_1 = a_0 \cdot b_1 \oplus a_1 \cdot b_0 \oplus a_2 \cdot b_3 \oplus a_3 \cdot b_2 \oplus a_1 \cdot b_3 \oplus a_2 \cdot b_2 \oplus a_3 \cdot b_1; \\ c_0 = a_0 \cdot b_0 \oplus a_1 \cdot b_3 \oplus a_2 \cdot b_2 \oplus a_3 \cdot b_1. \end{cases}$$

Таким образом, мы имеем m аддитивных функций для вычисления коэффициентов $c_{m-1}\dots c_0$. Функции содержат слагаемые в виде произведений коэффициентов $a_i \cdot b_j$, где $i, j = 0\dots m-1$. Поскольку мы имеем дело с полями $GF(2^m)$, являющиеся расширением базового простого поля $GF(2)$, то произведение коэффициентов эквивалентно логическому умножению (конъюнкции).

Для аппаратной реализации таких функции удобнее использовать специализированные программируемые логические матрицы (ПЛМ), содержащие в себе логические элементы «И» с двумя входами и многовходовые сумматоры по модулю 2.

Ниже на рис. 1 приведена функциональная схема умножителя элементов поля $GF(2^4)$, образованного на базе примитивного неприводимого многочлена $p(x) = x^4 + x + 1$.

Входы сумматоров в соответствии с аддитивными функциями подключаются к выходам логических элементов «И», формирующих соответствующие произведения коэффициентов $a_i \cdot b_j$. Незадействованные входы сумматоров подключаются к «земле».

Теперь обобщим вышеприведенный пример для общего случая умножения элементов поля Галуа $GF(2^m)$, $m \geq 2$, образованного на базе заданного примитивного неприводимого многочлена $p(x) = x^m + p_{m-1}x^{m-1} + p_1 x + p_0$, в виде итерационной процедуры, в которой за m итераций выводятся m формул для расчета всех коэффициентов многочлена-произведения:

$$
\begin{aligned}
&(a(x)\cdot b(x)) \bmod (p(x)) = \sum_{k=0}^{m-1} c_k^{(m)} \cdot x^k;\\
&s = 1\dots m; \quad m \geq 2; \quad c_0^{(0)} = \dots = c_{m-1}^{(0)} = 0;\\
&\begin{cases}
c_0^{(s)} = \quad c_{m-1}^{(s-1)} \cdot p_0 \oplus a_0 \cdot b_{m-s};\\
c_1^{(s)} = c_0^{(s-1)} \oplus c_{m-1}^{(s-1)} \cdot p_1 \oplus a_1 \cdot b_{m-s};\\
\vdots\\
c_{m-1}^{(s)} = c_{m-2}^{(s-1)} \oplus c_{m-1}^{(s-1)} \cdot p_{m-1} \oplus a_{m-1} \cdot b_{m-s}.
\end{cases}
\end{aligned}
\qquad (2)
$$

Следует особо отметить, что итерационный вывод прямых формул осуществляется лишь один раз на этапе проектирования аппаратной реализации, и далее формулы аппаратно реализуются в коммутационной матрице соединений между выходами m^2 двухвходовых элементов «И» и входы m элементов сумматоров по модулю 2.

Что касается операции деления элемента a на ненулевой элемент b поля, то ее можно свести к операции умножения на обратный элемент b^{-1}.

$$
\forall a, b \in GF(2^m) : b \neq 0 \Rightarrow \underbrace{a / b}_{GF(2^m)} = \underbrace{a \cdot b^{-1}}_{GF(2^m)}. \qquad (3)
$$

Вычисление обратного элемента по умножению для максимального быстродействия, очевидно, лучше всего опять же осуществлять табличным способом, используя ПЗУ емкостью $2^m m$ бит. Что касается формирования самой таблицы обратных элементов на этапе проектирования, то ее можно подготовить заранее, используя расширенный алгоритм Евклида для многочленов, который сводит решение уравнения

$$
\underbrace{(b(x)\cdot b^{-1}(x)) \bmod p(x) = 1}_{GF(2)},
$$

где $b^{-1}(x)$ разыскиваемый обратный многочлен по умножению, к нахождению многочленов $g(x)$ и $h(x)$, а также наибольшего общего делителя $НОД(b(x), p(x))$ для многочленов $b(x)$ и $p(x)$ таких, что

$$
\underbrace{b(x)\cdot g(x) + p(x)\cdot h(x) = НОД(b(x), p(x))}_{GF(2)}.
$$

Поскольку мы имеем дело с полем, то для любого ненулевого многочлена $b(x)$ алгоритм в качестве $НОД(b(x), p(x)$ дает скаляр $\lambda \in GF(2)$ (многочлен нулевой степени), и в нашем «двоичном» случае скаляр будет равен строго $\lambda = 1$, так как в базовом простом поле $GF(2)$ существует только один ненулевой элемент – это единица. Соответственно, многочлен $g(x)$, находимый алгоритмом, и является искомым обратным многочленом по умножению, то есть $b^{-1}(x) = g(x)$.

Приведем для примера таблицу обратных элементов для элементов поля $GF(2^4)$, представленных для компактности в виде десятичных и двоичных эквивалентов элементов.

$(b)_{10}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$(b)_2$	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
$(b^{-1})_{10}$	1	9	14	13	11	7	6	15	2	12	5	10	4	3	8
$(b^{-1})_2$	0001	1001	1110	1101	1011	0111	0110	1111	0010	1100	0101	1010	0100	0011	1000

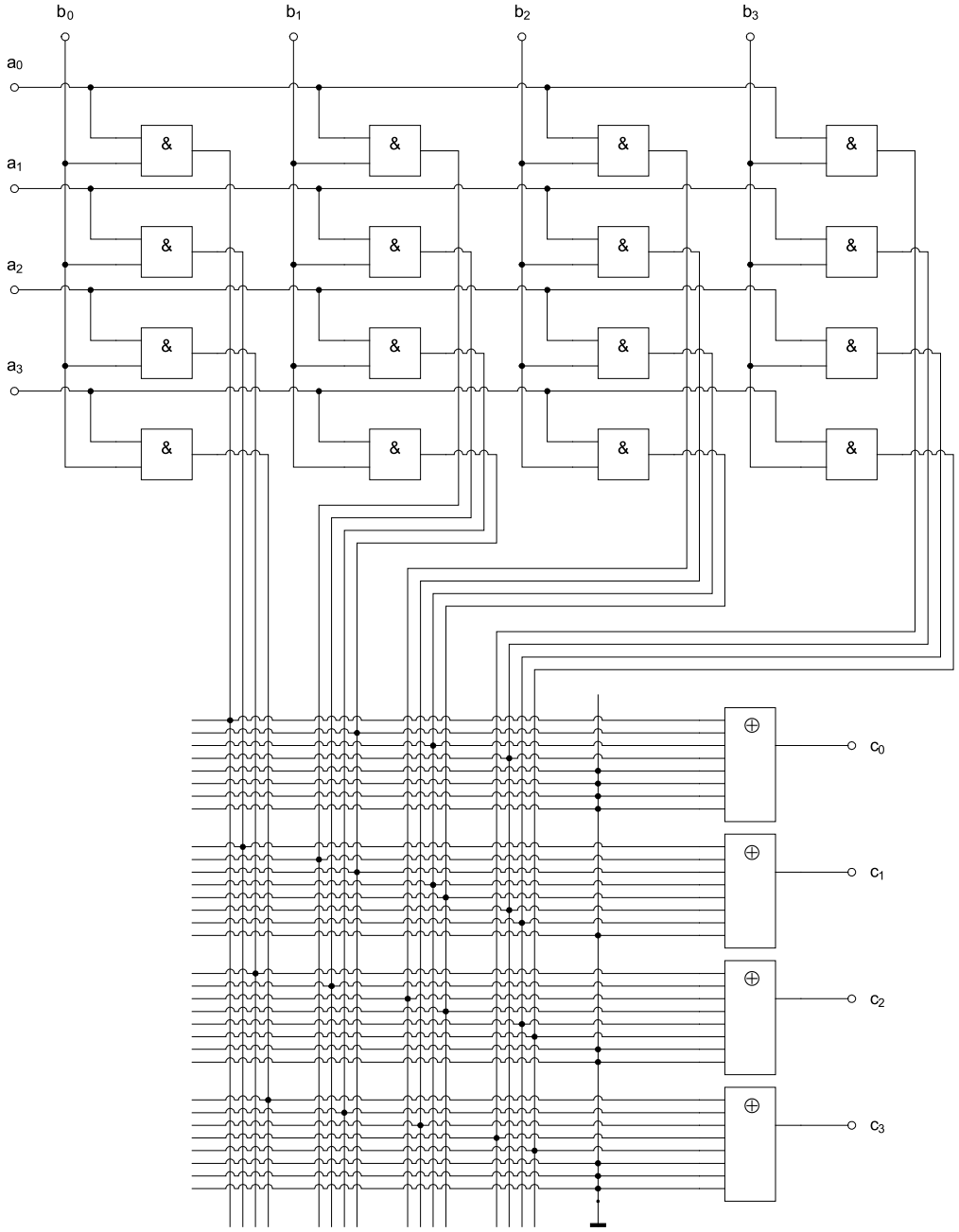

Рис. 1. Функциональная схема умножителя элементов поля Галуа GF(2⁴) на базе специализированной логической матрицы

Арифметический процессор для поля Галуа *GF*(2^m) на базе быстрого умножения и инвертирования элементов. Используя умножитель элементов поля Галуа *GF*(2^m) на базе специализированной программируемой логической матрицы теперь можно построить арифметический процессор для поля Галуа *GF*(2^m), сведя операцию деления элемента a на ненулевой элемент b поля к умножению на обратный элемент b^{-1} по умножению. Вычисление обратного элемента по умножению для максимального быстродействия, очевидно, лучше осуществлять табличным способом, используя ПЗУ емкостью $2^m m$ бит.

Ниже на рисунке 2 представлена функциональная схема арифметического процессора для поля Галуа *GF*(2^m), использующего таблицу обратных элементов по умножению, хранящуюся в ПЗУ и умножитель элементов, который, как было рассмотрено выше, реализуется на базе специализированной программируемой логической матрицы.

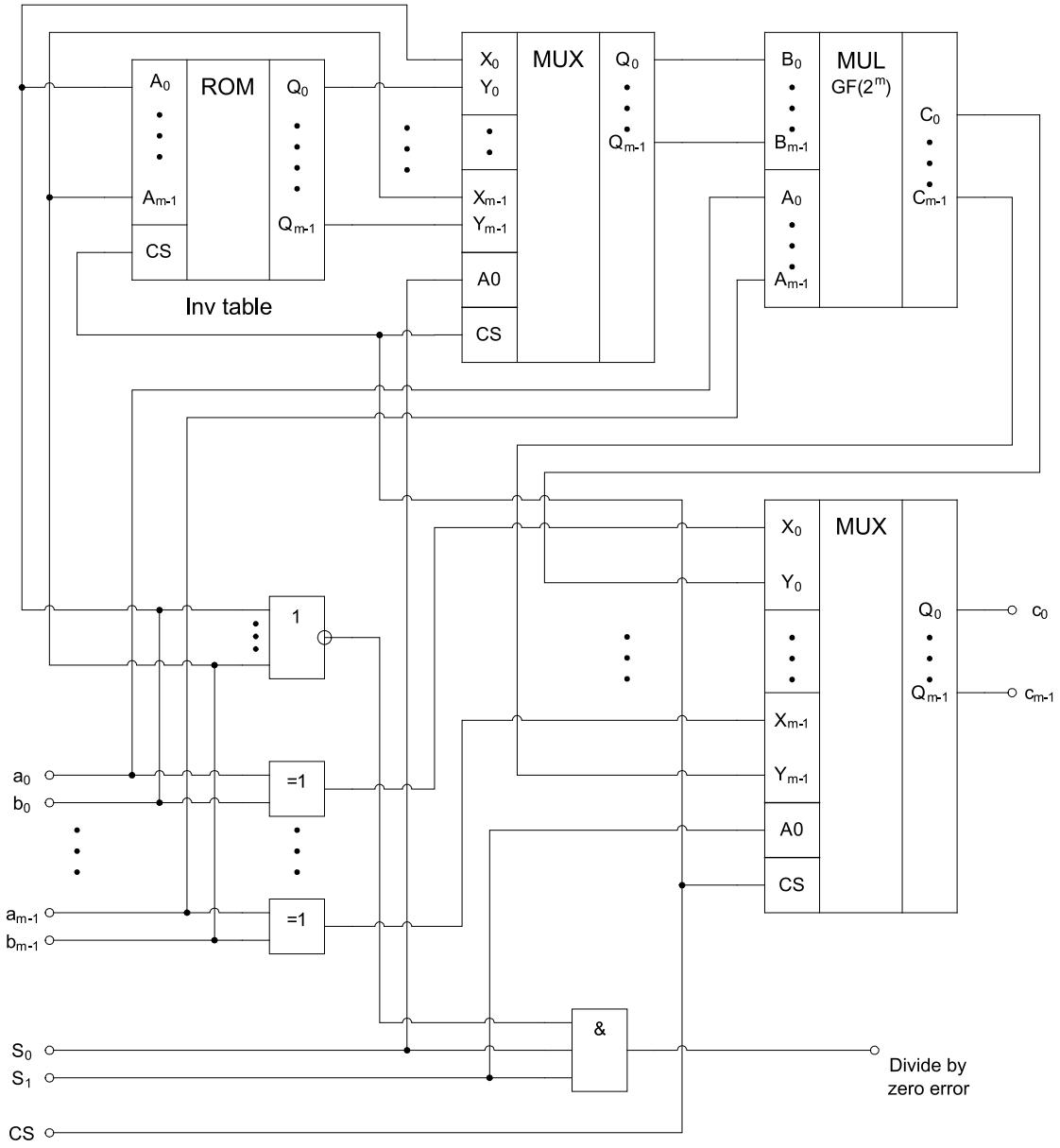

Рис. 2. Функциональная схема арифметического процессора для поля Галуа GF(2^m) с использованием умножителя элементов и таблицы обратных элементов по умножению

В схеме процессора используются два m-битных мультиплексора 2 → 1. Нижний мультиплексор коммутирует свои входы с выходами логических элементов XOR при управляющем сигнале S1 = 0 (режим операций сложения / вычитания), и с выходами умножителя при S1 = 1 (режим операций умножения / деления). При S1 = 0, управляющий сигнал S0 не играет никакой роли (сложение и вычитание сводится к одной и той же операции «побитового» XOR). При S1 = 1, сигнал S0 управляет верхним мультиплексором, который при S0 = 0 подключает линии $b_{m-1}\ldots b_0$ напрямую к умножителю элементов, что соответствует операции умножения на операнд b, а при S0 = 1 подключает выходы ПЗУ, преобразующего операнд b в его обратный элемент по умножению, что соответствует операции деления на операнд b. В схеме процессора также предусмотрена цепь обнаружения нулевого делителя (b = 0) в режиме операции деления (S1 = 1 и S0 = 1).

Арифметический процессор на базе умножителя и таблицы обратных элементов требует одного ПЗУ емкостью $2^m m$ бит, и умножителя, состоящего из m^2 двухвходовых логических элементов «И», m многовходо-

вых сумматоров по модулю 2, и коммутационной матрицы размером $(m^2+1)m^2m \sim m^5$. Если коммутационную матрицу тоже рассматривать как «постоянную память», то, очевидно, что ее размер составляет m^5.

Заключение

Таким образом, в данной статье рассмотрены поля Галуа $GF(2^m)$, их арифметика, операции сложения, а также умножения и деления на базе быстрого умножения и табличного инвертирования элементов. Также рассмотрена схема быстрого умножения, а также предлагаемая автором схема арифметического процессора для поля Галуа.

Полученные результаты были использованы автором для разработки обучающей программы и лабораторных стендов для изучения студентами технических специальностей технологии кодирования информации при применении кодов Рида-Соломона.

Список литературы

1. Todd K. Moon. Error correcting coding: mathematical methods and algorithms. – Hoboken, New Jersey: John Wiley & Sons Inc., 2005.

2. Рахман П.А., Григорьева Т.В. Кодирование информации с применением кодов Рида-Соломона. – Уфа: Изд-во УГНТУ, 2015.

3. Рахман П.А., Шарипов М.И. Модель надежности двухузлового кластера приложений высокой готовности в системах управления предприятием // Экономика и менеджмент систем управления. – 2015. – Т. 17, № 3. – С. 85–102.

4. Рахман П.А., Шарипов М.И. Модели надежности каскадных дисковых массивов в системах управления предприятием // Экономика и менеджмент систем управления. – 2015. – Т. 17. № 3.1. – С. 155–168.

5. Рахман П.А. Коэффициент готовности трехуровневых локальных сетей передачи данных // Международный журнал прикладных и фундаментальных исследований. – 2015. – № 9–3. – С. 463–466.

6. Рахман П.А. Показатели надежности восстанавливаемых систем с заданным порогом аварийного отключения // Международный журнал прикладных и фундаментальных исследований. – 2015. – № 9–3. – С. 467–470.

7. Рахман П.А. Среднее время до потери данных двухдискового массива // Международный журнал прикладных и фундаментальных исследований. – 2015. – № 9–4. – С. 603–607.

8. Рахман П.А. Коэффициент готовности системы обработки данных с основным и резервным узлами // Международный журнал прикладных и фундаментальных исследований. – 2015. – № 9–4. – С. 608–611.

9. Рахман П.А. Модель надежности мажоритарной вычислительной системы на базе элементов с тремя состояниями // Международный журнал прикладных и фундаментальных исследований. – 2015. – № 10–1. – С. 33–37.

10. Рахман П.А. Алгоритм выбора кратности исправляемых искажений для кодирования информации с применением кодов Рида-Соломона // Международный журнал прикладных и фундаментальных исследований. – 2015. – № 10–2. – С. 208–212.