УДК 004.047 DOI: 10.31114/2078-7707-2022-4-136-141

Исследование корректирующей способности модулярных кодов, применяемых в AES-системах

И.А. Проворнов

Северо-Кавказский федеральный университет, г. Ставрополь, igorprovornov@yandex.ru

***Аннотация* — В настоящее время предложены методы повышения надежности SPN-криптосистем, основанные на применение полиноминальной системы классов вычетов. Применение подобных кодов позволяет перенести вычисления из области $GF(2^8)$ в область $GF(2^4)$, что положительно отражается на скорости и надежности функционирования вычислительных систем. В данной статье описываются результаты применения подобных технологий в процедурах SubBytes и MixColumns SPN-систем, исследуется эффективность корректирующей способности подобных кодов, оценивается выигрыш относительно классического варианта построения криптосистем.**

***Ключевые слова* — SPN-криптосистемы, надежность, модулярная арифметика, поля Галуа.**

I. Введение

Известно, что в настоящее время к надежности криптографических систем предъявляются высокие требования надежности и безопасности. Большинство современных криптографических систем в качестве элементной базы используют различные микроконтроллеры или микропроцессоры, которые как правило реализованы с помощью систем на кристалле (СнК). Вместе с тем, подобные системы подвержены различным типам атак, в том числе возможных за счет аппаратных сбоев в работе шифратора. Сбои могут быть вызваны как низкой надёжностью элементов, входящих в состав системы, так условиями эксплуатации, непредусмотренными для данных устройств (низкие температуры, высокий уровень радиации и т.п.) [1].

По состоянию на сегодняшний день криптоалгоритм AES находит широкое применение в информационно-телекоммуникационных системах, что обусловлено следующими его преимуществами:

1) Высокая криптостойкость.

2) Высокая скорость работы за счет возможности реализации параллельных вычислений.

3) Байт-ориентированность, повышающая эффективность реализации в компьютерных системах.

В свою очередь, такое широкое распространение криптоалгоритма AES спровоцировало большое количество атак, в том числе на основе сбоев в работе шифраторов.

Поэтому являются актуальными задачи:

1) разработка математических методов повышения надежности выполнения алгоритма;

2) оценка корректирующей способности разработанных методов;

3) разработка структурной схемы системы, реализующей данные методы.

II. Анализ принципов построения и функционирования SPN-систем

На вход шифратора SPN-системы поступает блок открытой информации размером 16 байт (128 бит), при этом если объем текста не кратен данному значению происходит автоматическое добавление символов. Блок данных в алгоритме AES именуется *state* и обычно визуально представляется в виде матрицы размером 4x4 (рис.1).

S_{00}	S_{01}	S_{02}	S_{03}
S_{10}	S_{11}	S_{12}	S_{13}
S_{20}	S_{21}	S_{22}	S_{23}
S_{30}	S_{31}	S_{32}	S_{33}

Рис. 1. Представление входного блока SPN-системы

Далее каждый блок подвергается нескольким раундам шифрования, которые в свои очередь могут состоять из процедур:

1) SubBytes (замена байтов в соответствии с заранее определенной таблицей).

2) ShiftRows (циклический сдвиг строки на определенное количество позиций).

3) MixColumns (перемешивание байтов внутри столбца.

4) AddRoundKey (суммирование с раундовоым ключом).

Расширенные ключ состоит из 44 слов w_i, где w_{0-3} – исходный ключ, w_{4-44} – совокупность раундовых ключей, генерируемых по правилам:

если кратно w_i кратно 4:

$$w_i = SubBytes(RotByte(w_{i-1})) \otimes Rcon_{(i/4)},$$

если w_i *кратно 4* не кратно 4:

$$w_i = w_{i-4} \otimes_{wi-1} .$$

При этом состав раунда зависит от его номера в алгоритме (рис. 2).

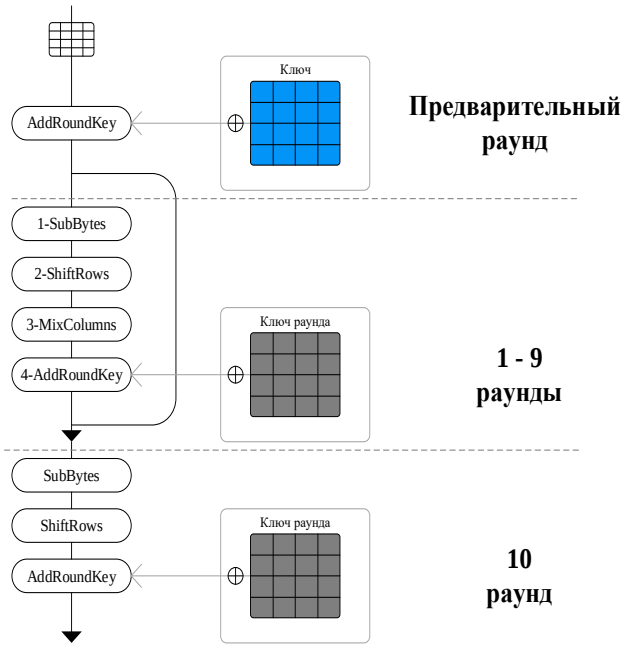

Рис. 2. Состав раундов AES

Раунды 1 – 9 идентичны и представляют собой последовательное выполнение операций SubBytes, ShiftRows, MixColumns и AddRoundKey. 10 раунд отличается отсутствием процедуры MixColumns.

Таким образом, задачу повышения надежности SPN-криптосистемы можно свести к повышению надежности выполнения данных процедур. Кроме того предлагается ввести допущение, что ошибки возникают при выполнении математических операции сложения и умножения. Тогда остается необходимость модернизации процедур SubBytes и MixColumns [3].

III. Используемый математический аппарат

В классической реализации процедуры SubBytes предлагается использовать таблицу замены 16x16 (рис. 3). Операция замены проводится независимо для каждого бита из входного массива state.

hex	y															
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**a**	**b**	**c**	**d**	**e**	**f**
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	c5	7b	f2	63
1	ca	82	c9	7d	be	45	e9	af	f7	d4	a2	da	af	7d	be	ca
2	b7	fd	93	26	c4	f4	89	be	ae	a5	e5	de	be	26	c4	b7
3	04	da	23	c3	4	4d	1a	34	85	12	50	5b	34	c3	4	04
4	09	fe	2c	1a	a6	d7	4f	b3	ed	b3	f5	a2	b3	1a	a6	09
5	0	4f	00	ed	aa	34	4b	45	38	82	1f	e8	45	ed	aa	0
6	d0	7a	aa	fb	a3	b4	7a	1e	01	e5	00	b2	1e	fb	a3	d0
7	51	12	40	1b	4f	d6	78	f4	2e	a6	52	e9	f4	1b	4f	51
8	0	65	13	33	bf	3a	55	78	4b	f9	fa	a7	78	33	bf	0
9	cd	0a	4f	85	ba	33	00	43	6a	33	b5	7f	43	85	ba	cd
a	60	5b	3a	9f	78	a5	e1	28	1f	a7	e8	2a	28	9f	78	60
b	e0	8c	37	8f	e1	b4	4b	e3	8a	5b	5a	5b	e3	8f	e1	e0
c	e7	cc	25	38	2e	f6	5a	e6	b7	1e	54	bb	e6	38	2e	e7
d	ba	da	b5	24	bd	65	a6	b6	07	8a	f9	b9	b6	24	bd	ba
e	70	df	98	0e	d6	b2	89	60	01	56	10	39	60	0e	d6	70
f	e1	b1	89	2d	e1	2a	0e	01	a5	00	0a	0b	01	2d	e1	e1

x

Рис. 3. Классическая таблица SubBytes

Однако такой подход не предоставляет возможность проводить коррекцию ошибок, возникших в результате сбоев работы шифратора при реализации процедуры SubBytes. Поэтому предлагается перейти к использованию полиноминальной системы классов вычетов (ПСКВ).

Для этого каждый входной бит переводится в полиноминальную систему классов вычетов с основаниями $p_1(z) = z^4 + z + 1$ и $p_2(z) = z^4 + z^3 + 1$:

$$A = (\alpha_1(x), \alpha_2(x),) .$$

Таким образом, представление бита для дальнейших операций изменяется с 8-разрядного числа на два 4-разрядных, что открывает дополнительные возможности за счет применения технологий параллельных вычислений [2]. Однако, вместо одной таблицы необходимо использовать 2, а в случае использования механизмов коррекции – 4. При этом дополнительные (контрольные) основания рассчитываться следующим образом:

$$\alpha_3(x) = \alpha_1(x) + \alpha_2(x), \quad (1)$$

$$\alpha_4(x) = (\alpha_1 + x \cdot \alpha_2(x)) \bmod p_3(x) . \quad (2)$$

В качестве третьего контрольного основания предлагается использовать полином $p_3(z) = z^4 + z^3 + z^2 + z + 1$.

При построении таблиц S-блока в качестве строк таблиц используют α_1, в качестве столбцов – α_2.

Вместе с тем, для обеспечения интеграции результатов процедуры SubBytes, реализованной в ПСКВ в SPN-системы, построенные по стандартному принципу, необходимо таблицы замен проектировать таким образом, чтобы выходные значения процедуры SubBytes были идентичны классическому алгоритму.

Пусть на вход S-блока поступает число 19_{16}. В классической таблице S блока ему соответствует число $D4_{16}$, соответственно это же число должно быть на выходе S-блока, реализованного в ПСКВ. Для построения таблиц необходимо выполнить перевод входного бита в ПСКВ:

$$\alpha_1(19_{16}) = 19_{16} \bmod p_1 = A ,$$

$$\alpha_2(19_{16}) = 19_{16} \bmod p_2 = 0 .$$

Для определения содержимого таблиц, которое будет располагаться на пересечении данных значений необходимо выходное значение классического алгоритма перевести в ПСКВ:

$$\alpha_1(D4_{16}) = D4_{16} \bmod p_1 = 0 ,$$

$$\alpha_2(D4_{16}) = D4_{16} \bmod p_2 = 5 ,$$

$$\alpha_3(D4_{16}) = \alpha_1 + \alpha_2 = 5 ,$$

$$\alpha_4(D4_{16}) = (\alpha_1 + (\alpha_2 * z)) \bmod p_3 = A .$$

Для построения 4 таблиц по этому принципу необходимо провести вышерассмотренные преобразования для всех числе в GF(2^8). Результаты

данных расчетов представлены на рисунках Рис. 4 – Рис. 7.

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	9	9	2	f	d	b	b	5	b	f	0	1	4	3	f	9
1	d	5	9	0	9	3	4	f	4	f	6	0	4	8	9	1
2	a	2	e	5	a	4	6	c	2	4	d	6	6	d	f	8
3	f	0	0	2	0	1	b	1	2	f	e	1	c	e	1	0
4	8	3	6	5	0	5	c	d	d	6	4	7	5	0	9	f
5	4	2	2	7	7	1	2	5	4	0	3	e	c	5	6	4
6	a	b	7	e	d	e	5	1	7	0	c	c	7	7	e	c
7	e	f	4	e	d	3	8	2	1	9	8	a	2	b	5	b
8	2	7	9	3	a	0	f	4	5	5	8	d	5	0	8	b
9	8	d	f	d	b	a	1	9	6	1	5	c	b	a	3	3
a	4	8	7	6	7	3	2	f	d	4	d	2	4	7	7	7
b	3	e	0	c	8	5	9	e	8	3	3	d	1	c	8	d
c	8	b	9	f	e	0	6	9	e	6	3	c	c	1	b	9
d	5	7	b	b	c	2	6	8	d	e	c	8	7	3	a	e
e	2	a	a	a	2	b	1	0	a	a	4	6	a	8	6	f
f	1	6	3	3	6	b	c	7	1	a	e	c	9	9	a	f

Рис. 4. Таблица вычисления остатка α_1 блока выходного значения S-блока

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	7	f	8	3	5	c	b	8	6	4	5	4	e	b	c	b
1	b	1	4	6	d	9	b	a	d	f	a	1	6	b	c	2
2	2	0	a	7	f	4	3	2	3	5	e	b	1	9	9	9
3	2	2	4	6	f	8	0	9	1	e	6	7	6	2	6	9
4	3	3	d	5	8	a	a	6	f	c	1	4	e	0	8	0
5	7	9	4	d	2	f	7	9	3	a	8	c	e	f	4	2
6	9	1	a	e	8	9	b	0	6	c	c	9	8	9	5	5
7	8	d	c	b	4	d	6	d	d	1	c	5	5	f	0	2
8	e	e	5	7	e	d	6	f	2	6	a	c	3	7	e	7
9	7	d	1	2	d	0	5	e	6	1	c	0	5	c	1	f
a	0	8	b	5	1	e	b	7	7	a	3	f	9	7	f	c
b	2	1	3	d	5	4	2	0	2	5	a	0	b	1	1	a
c	f	4	a	5	d	b	f	0	f	9	0	7	4	3	e	3
d	d	3	a	3	b	c	8	0	1	7	8	d	0	6	7	3
e	2	1	b	d	a	9	a	e	4	8	8	0	3	4	7	8
f	e	2	4	c	e	8	3	5	c	6	4	f	9	6	a	b

Рис. 5. Таблица вычисления остатка α_2 блока выходного значения S-блока

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	e	6	a	c	8	7	0	d	d	b	5	5	a	8	3	2
1	6	4	d	6	4	a	f	5	9	0	c	1	2	3	5	3
2	8	2	4	2	5	0	5	e	1	1	3	d	7	4	6	1
3	d	2	4	4	f	9	b	8	3	1	8	6	a	c	7	9
4	b	0	b	0	8	f	6	b	2	a	5	3	b	0	1	f
5	3	b	6	a	5	e	5	c	7	a	b	2	2	a	2	6
6	3	a	d	0	5	7	e	1	1	c	0	5	f	e	b	9
7	6	2	8	5	9	e	e	f	c	8	4	f	7	4	5	9
8	c	9	c	4	4	d	9	b	7	3	2	1	6	7	6	c
9	f	0	e	f	6	a	4	7	0	0	9	c	e	6	2	c
a	4	0	c	3	6	d	9	8	a	e	e	d	d	0	8	b
b	1	f	3	1	d	1	b	e	a	6	9	d	a	d	9	7
c	7	f	3	a	3	b	9	9	1	f	3	b	8	2	5	a
d	8	4	1	8	7	e	e	8	c	9	4	5	7	5	d	d
e	0	b	1	7	8	2	b	e	e	2	c	6	9	c	1	7
f	f	4	7	f	8	3	f	2	d	c	a	3	0	f	0	4

Рис. 6. Таблица вычисления остатка α_3 блока выходного значения S-блока

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	7	8	d	9	7	c	2	a	7	7	a	9	7	a	8	0
1	4	7	1	c	c	e	d	4	1	e	d	2	8	1	e	5
2	e	2	5	b	b	c	0	8	4	e	e	f	4	0	2	5
3	b	4	8	e	1	e	b	c	0	c	2	f	0	a	d	d
4	e	5	3	f	f	e	7	1	c	1	6	f	6	0	6	f
5	a	f	a	2	3	0	c	8	2	b	c	9	f	4	e	0
6	7	9	c	d	2	3	c	1	b	7	b	1	8	a	4	6
7	1	a	3	7	5	6	4	7	4	b	f	0	8	a	5	f
8	1	4	3	d	9	5	3	5	1	9	3	a	3	e	b	5
9	6	8	d	9	e	a	b	a	a	3	2	c	1	d	1	2
a	4	7	e	c	5	0	b	1	3	f	b	3	9	9	6	0
b	7	c	6	9	2	d	d	e	c	9	8	d	8	e	a	6
c	9	3	2	5	b	9	7	9	f	b	3	2	4	7	8	f
d	0	1	0	d	5	5	9	8	f	0	3	d	7	f	4	8
e	6	8	3	f	9	6	a	3	2	5	b	6	c	0	8	0
f	2	2	b	4	5	4	a	d	6	6	6	d	4	5	1	6

Рис. 7. Таблица вычисления остатка α_3 блока выходного значения S-блока

Аналогичный математический аппарат используется при построении таблиц для преобразования MixColumns. В ходе преобразования MixColumns выполняется умножение полиномов на константу 2_{16}, в ходе которого может случиться сбой

работы шифратора под действием различных внутренних и внешний факторов. Вместе с тем, результат умножения любого числа из множества $GF(2^8)$ на константу 2_{16} известен, поэтому рационально проводить эту операцию в табличном виде. Для построения необходимый таблиц необходимо для каждого элемента $S_{вх}$ из $GF(2^8)$:

1) Вычислить $\alpha_1(S_{вх})$:

$$\alpha_1(S_{вх}) = S_{вх} \bmod p_1.$$

2) Вычислить $\alpha_2(S_{вх})$:

$$\alpha_2(S_{вх}) = S_{вх} \bmod p_2.$$

3) Вычислить $S_{вых}$:

$$S_{вых}=S_{вх}*2_{16}.$$

4) Вычислить $\alpha_1(S_{вых})$:

$$\alpha_1(S_{вых}) = S_{вых} \bmod p_1.$$

5) Вычислить $\alpha_2(S_{вых})$.

$$\alpha_2(S_{вых}) = S_{вых} \bmod p_2$$

6) Вычислить $\alpha_3(S_{вых})$ (формула 1).

7) Вычислить $\alpha_4(S_{вых})$ (формула 2).

На основе полученных данных строятся таблицы, используемые для реализации операции умножения на константу 2_{16} в ходе процедуры MixColumns. Данные таблицы позволяют избежать ошибок при выполнении умножения в модулярных кодах за счет введения избыточности в виде дополнительного основания ПСКВ.

Данные таблицы представлены на рисунках 8 – 11.

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	0	f	9	6	8	7	1	e	e	1	7	8	6	9	f	0
1	d	2	4	b	5	a	c	3	3	c	a	5	b	4	2	d
2	d	2	4	b	5	a	c	3	3	c	a	5	b	4	2	d
3	0	f	9	6	8	7	1	e	e	1	7	8	6	9	f	0
4	0	f	9	6	8	7	1	e	e	1	7	8	6	9	f	0
5	d	2	4	b	5	a	c	3	3	c	a	5	b	4	2	d
6	d	2	4	b	5	a	c	3	3	c	a	5	b	4	2	d
7	0	f	9	6	8	7	1	e	e	1	7	8	6	9	f	0
8	d	2	4	b	5	a	c	3	3	c	a	5	b	4	2	d
9	0	f	9	6	8	7	1	e	e	1	7	8	6	9	f	0
a	0	f	9	6	8	7	1	e	e	1	7	8	6	9	f	0
b	d	2	4	b	5	a	c	3	3	c	a	5	b	4	2	d
c	d	2	4	b	5	a	c	3	3	c	a	5	b	4	2	d
d	0	f	9	6	8	7	1	e	e	1	7	8	6	9	f	0
e	0	f	9	6	8	7	1	e	e	1	7	8	6	9	f	0
f	d	2	4	b	5	a	c	3	3	c	a	5	b	4	2	d

Рис. 8. Таблица вычисления остатка α_1 блока умножения на 2_{16}

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	0	c	c	0	0	c	c	0	c	0	0	c	c	0	0	c
1	e	2	2	e	e	2	2	e	2	e	e	2	2	e	e	2
2	8	4	4	8	8	4	4	8	4	8	8	4	4	8	8	4
3	6	a	a	6	6	a	a	6	a	6	6	a	a	6	6	a
4	8	4	4	8	8	4	4	8	4	8	8	4	4	8	8	4
5	6	a	a	6	6	a	a	6	a	6	6	a	a	6	6	a
6	0	c	c	0	0	c	c	0	c	0	0	c	c	0	0	c
7	e	2	2	e	e	2	2	e	2	e	e	2	2	e	e	2
8	5	9	9	5	5	9	9	5	9	5	5	9	9	5	5	9
9	b	7	7	b	b	7	7	b	7	b	b	7	7	b	b	7
a	d	1	1	d	d	1	1	d	1	d	d	1	1	d	d	1
b	3	f	f	3	3	f	f	3	f	3	3	f	f	3	3	f
c	d	1	1	d	d	1	1	d	1	d	d	1	1	d	d	1
d	3	f	f	3	3	f	f	3	f	3	3	f	f	3	3	f
e	5	9	9	5	5	9	9	5	9	5	5	9	9	5	5	9
f	b	7	7	b	b	7	7	b	7	b	b	7	7	b	b	7

Рис. 9. Таблица вычисления остатка α_2 блока умножения на 2_{16}

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	0	3	5	6	8	b	d	e	2	1	7	4	a	9	f	c
1	3	0	6	5	b	8	e	d	1	2	4	7	9	a	c	f
2	5	6	0	3	d	e	8	b	7	4	2	1	f	c	a	9
3	6	5	3	0	e	d	b	8	4	7	1	2	c	f	9	a
4	8	b	d	e	0	3	5	6	a	9	f	c	2	1	7	4
5	b	8	e	d	3	0	6	5	9	a	c	f	1	2	4	7
6	d	e	8	b	5	6	0	3	f	c	a	9	7	4	2	1
7	e	d	b	8	6	5	3	0	c	f	9	a	4	7	1	2
8	8	b	d	e	0	3	5	6	a	9	f	c	2	1	7	4
9	b	8	e	d	3	0	6	5	9	a	c	f	1	2	4	7
a	d	e	8	b	5	6	0	3	f	c	a	9	7	4	2	1
b	e	d	b	8	6	5	3	0	c	f	9	a	4	7	1	2
c	0	3	5	6	8	b	d	e	2	1	7	4	a	9	f	c
d	3	0	6	5	b	8	e	d	1	2	4	7	9	a	c	f
e	5	6	0	3	d	e	8	b	7	4	2	1	f	c	a	9
f	6	5	3	0	e	d	b	8	4	7	1	2	c	f	9	a

Рис. 10. Таблица вычисления остатка α_3 блока умножения на 2_{16}

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	0	8	e	6	8	0	6	e	9	1	7	f	1	9	f	7
1	e	6	0	8	6	e	8	0	7	f	9	1	f	7	1	9
2	2	a	c	4	a	2	4	c	b	3	5	d	3	b	d	5
3	c	4	2	a	4	c	a	2	5	d	b	3	d	5	3	b
4	f	7	1	9	7	f	9	1	6	e	8	0	e	6	0	8
5	1	9	f	7	9	1	7	f	8	0	6	e	0	8	e	6
6	d	5	3	b	5	d	b	3	4	c	a	2	c	4	2	a
7	3	b	d	5	b	3	5	d	a	2	4	c	2	a	c	4
8	7	f	9	1	f	7	1	9	e	6	0	8	6	e	8	0
9	9	1	7	f	1	9	f	7	0	8	e	6	8	0	6	e
a	5	d	b	3	d	5	3	b	c	4	2	a	4	c	a	2
b	b	3	5	d	3	b	d	5	2	a	c	4	a	2	4	c
c	8	0	6	e	0	8	e	6	1	9	f	7	9	1	7	f
d	6	e	8	0	e	6	0	8	f	7	1	9	7	f	9	1
e	a	2	4	c	2	a	c	4	3	b	d	5	b	3	5	d
f	4	c	a	2	c	4	2	a	d	5	3	b	5	d	b	3

Рис. 11. Таблица вычисления остатка α_4 блока умножения на 2_{16}

Кроме повышения скорости, проведение вычислений с использованием предложенного

математического аппарата позволяет проводить коррекцию ошибок, вызванных сбоями в работе шифратора. Для этого для контроля корректности проведения операции замены (умножения) вычисляются синдромы ошибки [4, 5]:

$$\delta_1(x) = \alpha_3(x) \oplus \alpha_3^*(x).$$

$$\delta_2(x) = \alpha 4(x) \oplus \alpha_4^*(x).$$

Синдромы ошибок однозначно указывают на местоположение ошибки в $S_{вых.}$

Проведенные исследования доказали возможность исправления однократных и двукратных ошибок.

IV. ОЦЕНКА ЭФФЕКТИВНОСТИ РАЗРАБОТАННЫХ АЛГОРИТМОВ

Вместе с тем существует необходимость сравнительного анализа показателей надежностей SPN-систем, реализованных классическим способом и с использованием ПСКВ.

В качестве показателя надёжности выберем вероятность безотказной работы, рассчитываемую по формуле:

$$P(t) = \frac{N_0 - n(t)}{N_0}$$

где N_0 –количество тактов работы шифраторов, находившихся по наблюдением,

n(t) –количество безаварийных тактов.

Известно, что при использовании метода непосредственного подсчета вероятностей с увеличением количества наблюдений статистическая оценка $P(t)$ приобретает свойство устойчивости, т.е. слабо отличается от вероятности безотказной работы.

В качестве модели потока отказов выберем простейший поток отказов, т.к. в рассматриваемой системе возникший отказ не увеличивает вероятность следующего отказа.

Для достижения подобного эффекта было выполнено моделирование 1000000 тактов работы шифратора SPN-системы.

Моделирование функционирования SPN-систем осуществлялось с помощью специального программного обеспечения (зарегистрировано в Реестре программ № 2022618470). В качестве эталонного значения вероятности безотказной работы выберем значение 99,9%. В качестве цели моделирования определим вычисление такого значения вероятности ошибки, при котором система перестанет соответствовать требованиям надежности.

Результаты моделирования представлены на рисунках 12, 13.

Рис. 12. Зависимость вероятности безотказной работы классической SPN-системы от вероятности сбоев

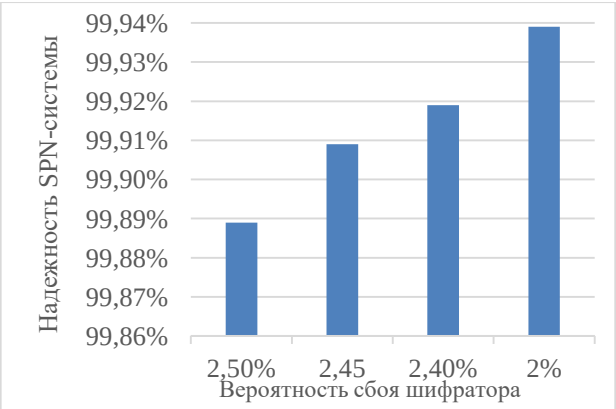

Рис. 13. Зависимость вероятности безотказной работы классической SPN-системы от вероятности сбоев

В результате установлено, что SPN-система, реализованная по классическим принципам, перестает соответствовать установленным требованиям надежности при достижении вероятности ошибки значения 0,02 %, а SPN-система, реализованная в ПСКВ – 2,5 %. Таким образом, по этому показателю надежности предлагаемый способ построения элементов криптосистем эффективнее существующих на 2,48 %.

V. ЗАКЛЮЧЕНИЕ

Таким образом, проведенные исследования доказывают целесообразность применения корректирующих кодов, реализованных в ПСКВ, в AES-криптосистемах. Тогда в свою очередь становится актуальной задача разработки схемотехнических решений, реализующий данную технологию.

ЛИТЕРАТУРА

[1] Червяков Н.И., Сахнюк П.А., Шапошников А.В., Ряднов С.А. Модулярные параллельные вычислительные структуры нейропроцессорных систем. М.: ФИЗМАТЛИТ, 203. – 288 с.

[2] Калмыков И.А., Гахов В.Р. и др. Применение корректирующих кодов полиноминальной системы классов вычетов для построения спецпроцессоров

цифровой обработки сигналов / Труды международного Форума по проблемам науки, техники и образования. Том 1./ Под ред.: В.П. Савиных, В.В. Вишневского. – М.: Академия наук, 2004. – С. 133-135.
[3] Калмыков И.А., Математические модели нейросетевых отказоустойчивых вычислительных средств, функционирующих в полиноминальной системе классов вычетов / Под ред. Н.И. Червякова. – М.: ФИЗМАТ-ЛИТ, 2005. – 276 с.
[4] Калмыков И.А., Степанова Е.П., Калмыков М.И., Топоркова Е.В. Применение корректирующих кодов полиноминальной системы классов вычетов для устранения последствий сбоев при шифровании алгоритмом AES \\ Международный журнал прикладных и фундаментальных исследований. – 2016. – №2-2. – С. 168-173.
[5] 2. Калмыков И.А., Степанова Е.П., Калмыков М.И., Топоркова Е.В., Повышение устойчивости к сбоям алгоритма шифрования AES на основе избыточной полиноминальной системы классов вычетов // Современные наукоемкие технологии. – 2015. – № 7. – С. 38-42.

Investigation of Corrective Ability of Modular Codes used in AES Systems

I. A. Provornov

North-Caucasus Federal University, Stavropol

***Abstract* — It is known that at present, high requirements are placed on the reliability of cryptographic systems (including SPN systems). As a rule, the fulfillment of these requirements is achieved through the implementation of methods that significantly increase circuit costs, which is their undoubted disadvantage. A promising direction for solving this problem is the introduction of corrective modular codes of the polynomial system of residue classes in SPN systems. The article analyzes the functioning of SPN-systems, defines their main components and procedures. The task of increasing the reliability of the SPN cryptosystem can be reduced to increasing the reliability of the execution of the SubBytes and MixColumns procedures. It is proposed to introduce the assumption that errors occur when performing mathematical operations of addition and multiplication. Then there remains the need to modernize the SubBytes and MixColumns procedures. The article discusses in detail the mathematical apparatus that makes it possible to implement error correction, including the reasons for replacing the polynomial used in the standard AES algorithm with alternative ones. The article describes the results of using modular polynomial codes in the SubBytes and MixColumns procedures of SPN systems, describes the principles of forming the tables used, investigates the effectiveness of the corrective ability of such codes, and evaluates the gain relative to the classical version of the construction of cryptosystems. When conducting research, the probability of failure-free operation is used as an indicator of reliability. It is proved that the use of corrective codes in a polynomial system of residue classes increases the overall reliability of the system. Development of circuit solutions for constructing a block for detecting and correcting errors in SPN cryptosystems has been indicated as a promising area of research.**

***Keywords* — SPN-cryptosystems, reliability, modular arithmetic, Galois fields.**

REFERENCES

[1] Hervyakov N.I., Sahnyuk P.A., SHaposhnikov A.V., Ryadnov S.A. Modular parallel computing structures of neuroprocessor systems. M.: FIZMATLIT, 2003. – 288 s.
[2] Kalmykov I.A., Gahov V.R. i dr. Application of corrective codes of the polynomial system of residue classes for the construction of special processors for digital signal processing / Trudy mezhdunarodnogo Foruma po problemam nauki, tekhniki i obrazovaniya. Tom 1./ Pod red.: V.P. Savinyh, V.V. Vishnevskogo. – M.: Akademiya nauk, 2004. – S. 133-135.
[3] Kalmykov I.A., Mathematical models of neural network fault-tolerant computing facilities operating in a polynomial system of residue classes / Pod red. N.I. CHervyakova. – M.: FIZMAT-LIT, 2005. – 276 s.
[4] Kalmykov I.A., Stepanova E.P., Kalmykov M.I., Toporkova E.V. . Application of correction codes of the polynomial system of residue classes to eliminate the consequences of failures in AES encryption \\ Mezhdunarodnyj zhurnal prikladnyh i fundamental'nyh issledovanij. – 2016. – №2-2. – S. 168-173.
[5] Kalmykov I.A., Stepanova E.P., Kalmykov M.I., Toporkova E.V. ., Increasing the failure resistance of the AES encryption algorithm based on the redundant polynomial system of residue classes \\ Mezhdunarodnyj zhurnal prikladnyh i fundamental'nyh issledovanij. – 2016. – №2-2. – S. 168-173.