Московский физико-технический институт
(государственный университет)
Физтех-школа радиотехники и компьютерных технологий
Кафедра информатики и вычислительной техники

Модули аппаратной поддержки криптографической защиты информации в ядре Эльбрус-v7

Студент: Зубрицкий И. В.

Научный руководитель: Альфонсо Д.М.

Москва 2023

Введение

- Симметричные шифры:
  - AES, ГОСТ 34.12-2018 (“Кузнечик”, “Магма”)
  - Мессенджеры защищают с помощью таких шифров переписку, а сервисы для видеосвязи — потоки аудио и видео.
- Ассиметричные шифры:
  - RSA, DSA
  - Асимметричные алгоритмы используют на стадии установки соединения.
- Алгоритмы хэширования:
  - SHA-256, ГОСТ 34.11-2018 (“Стрибог”)
  - Часто используется в алгоритмах электронно-цифровой подписи, в большинстве случаев вместо паролей хранятся значения их хеш-кодов.

Введение

Алгоритм	Время обработки блока (тактов)	
	Без аппаратной поддержки (Эльбрус-v5)	С аппаратной поддержкой (Intel) + AVX-512 (SkyLake)
“Кузнечик”	256	170
“Магма”	234	29
“Стрибог”	1124	432
“AES”	177	2.56
SHA	1844	115

Есть необходимость в аппаратной поддержке криптографии для новых МП «Эльбрус»

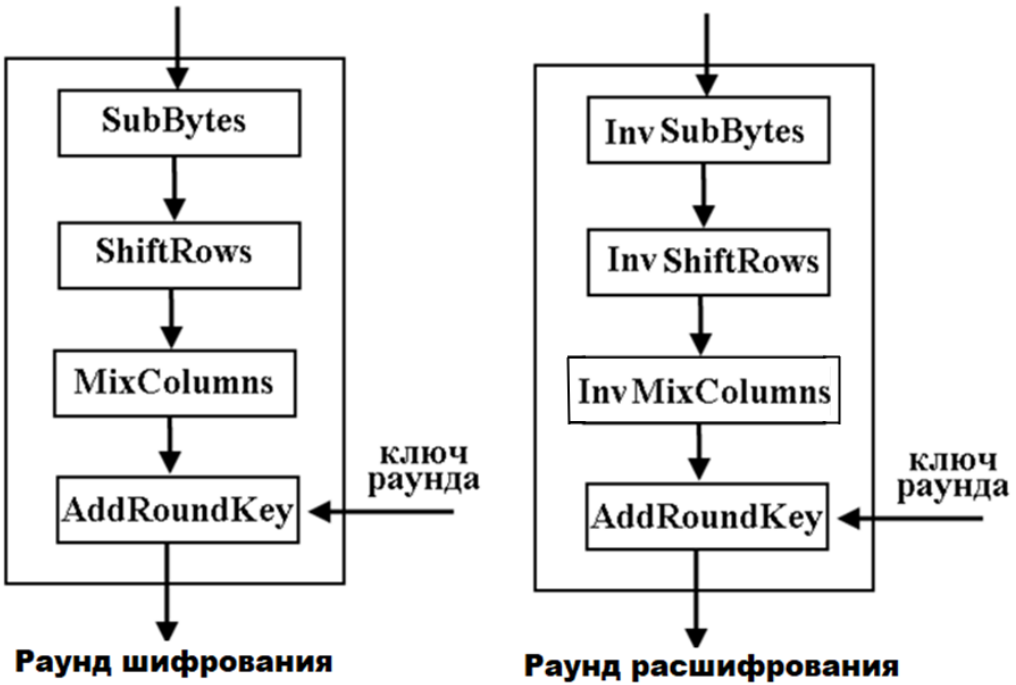
Цель работы

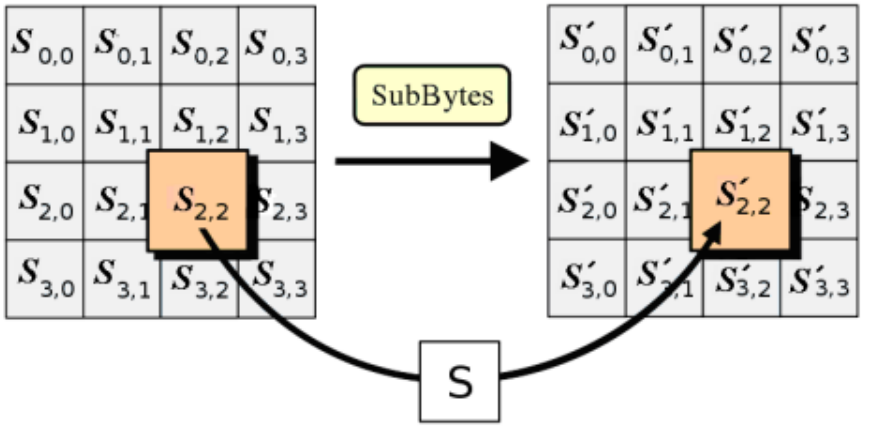
Разработать и отладить RTL-описания модулей аппаратного ускорения криптографических вычислений:

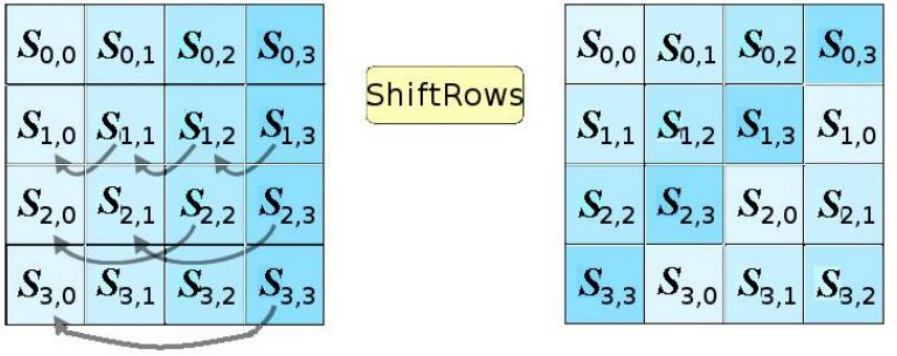
1. Зашифрование и расшифрование AES
2. Зашифрование “Магма”
3. Зашифрование “Кузнечик”
4. Вычисление хеш-функции “Стрибог”

AES (NIST FIPS 197)

Симметричный шифр с 10/12/14 типовыми раундами. Размер блока – 128 бит.







Линейное преобразование (MixColumns)

Действия в поле Галуа $GF(2^m)$:

1. Преобразование в многочлен:

   $01010111 = x^6 + x^4 + x^2 + x + 1$

2. Сложение <=> XOR

$(x^6 + x^4 + x^2 + x + 1) + (x^7 + x + 1) = x^7 + x^6 + x^4 + x^2$

3. Умножение: эквивалентно произведению полиномов с последующим нахождением остатка от деления на порождающий полином.

$(x^6 + x^4 + x^2 + x + 1)(x^7 + x + 1) = x^{13} + x^{11} + x^9 + x^8 + x^7 + x^7 + x^5 + x^3 + x^2 + x + x^6 + x^4 + x^2 + x + 1$

$x^{13} + x^{11} + x^9 + x^8 + x^6 + x^5 + x^4 + x^3 + 1 \bmod (x^8 + x^4 + x^3 + x + 1) = x^7 + x^6 + 1$

MixColumns:

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

InvMixColumns:

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

Линейное преобразование – аппаратная реализация

Примеры перехода от байтов к битовым матрицам:

$$1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad 2 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

InvMixColumns:

$$\begin{bmatrix} S'_{0,c} \\ S'_{1,c} \\ S'_{2,c} \\ S'_{3,c} \end{bmatrix} = \begin{pmatrix} a1 & 61 & 21 & e0 \\ e3 & a2 & 62 & 21 \\ c6 & 45 & c4 & 43 \\ 2d & eb & a9 & 67 \\ fa & b6 & 72 & 2e \\ f4 & 6c & e4 & 5c \\ e8 & d8 & c8 & b8 \\ d0 & b0 & 90 & 70 \\ 61 & 21 & e0 & a1 \\ a2 & 62 & 21 & e3 \\ 45 & c4 & 43 & c6 \\ eb & a9 & 67 & 2d \\ b6 & 72 & 2e & fa \\ 6c & e4 & 5c & f4 \\ d8 & c8 & b8 & e8 \\ b0 & 90 & 70 & d0 \\ 21 & e0 & a1 & 61 \\ 62 & 21 & e3 & a2 \\ c4 & 43 & c6 & 45 \\ a9 & 67 & 2d & eb \\ 72 & 2e & fa & b6 \\ e4 & 5c & f4 & 6c \\ c8 & b8 & e8 & d8 \\ 90 & 70 & d0 & b0 \\ e0 & a1 & 61 & 21 \\ 21 & e3 & a2 & 62 \\ 43 & c6 & 45 & c4 \\ 67 & 2d & eb & a9 \\ 2e & fa & b6 & 72 \\ 5c & f4 & 6c & e4 \\ b8 & e8 & d8 & c8 \\ 70 & d0 & b0 & 90 \end{pmatrix} \begin{bmatrix} S_{0,c} \\ S_{1,c} \\ S_{2,c} \\ S_{3,c} \end{bmatrix}$$

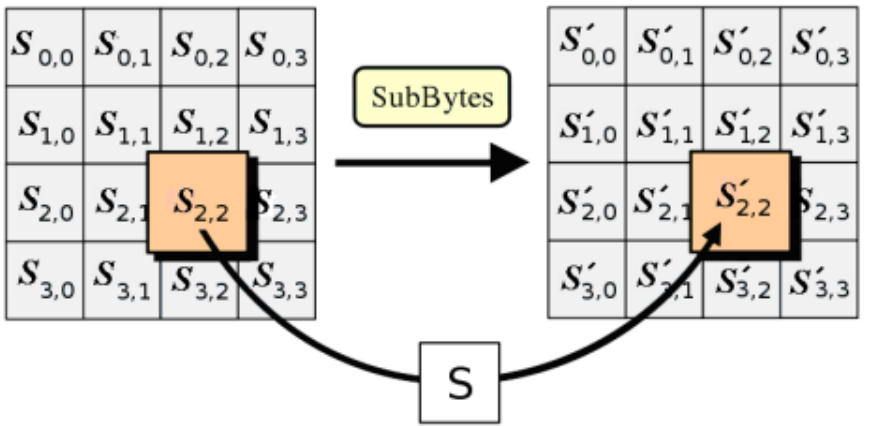
Таким образом матрица 4x4 байт представляются в виде матрицы 32x32 бит, умножения на которую является действием в поле GF(2), что **значительно проще для аппаратной реализации.**

Нелинейное преобразование (Sbox)

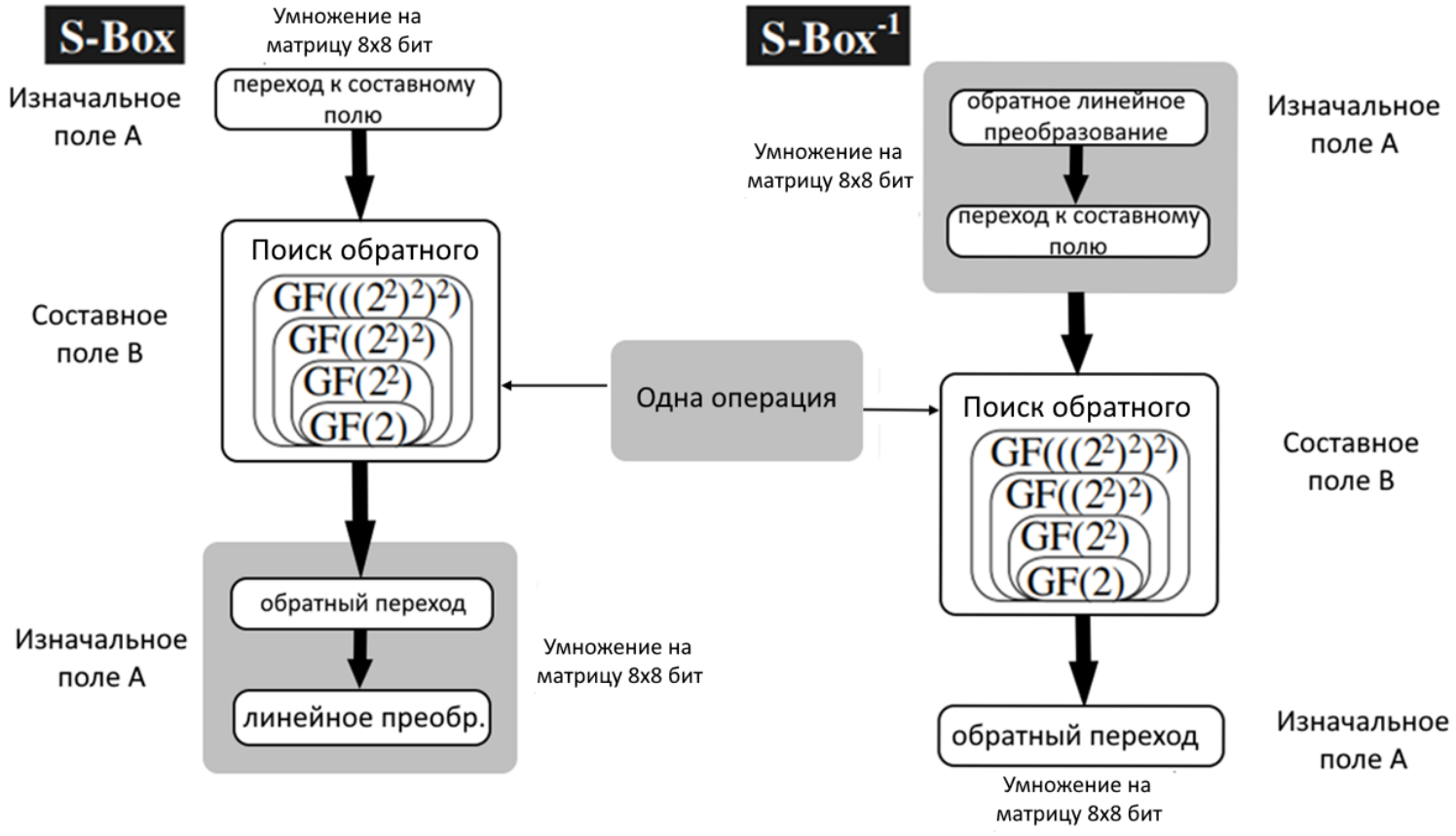
1. Нахождение обратного:
   $b' = b^{-1}$ в $GF(2^8)$ (если $b = 0$, то $b' = 0$)

2. Линейное преобразование байта:

$$\begin{bmatrix} b_0' \\ b_1' \\ b_2' \\ b_3' \\ b_4' \\ b_5' \\ b_6' \\ b_7' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}.$$



Оптимизация нелинейного преобразования

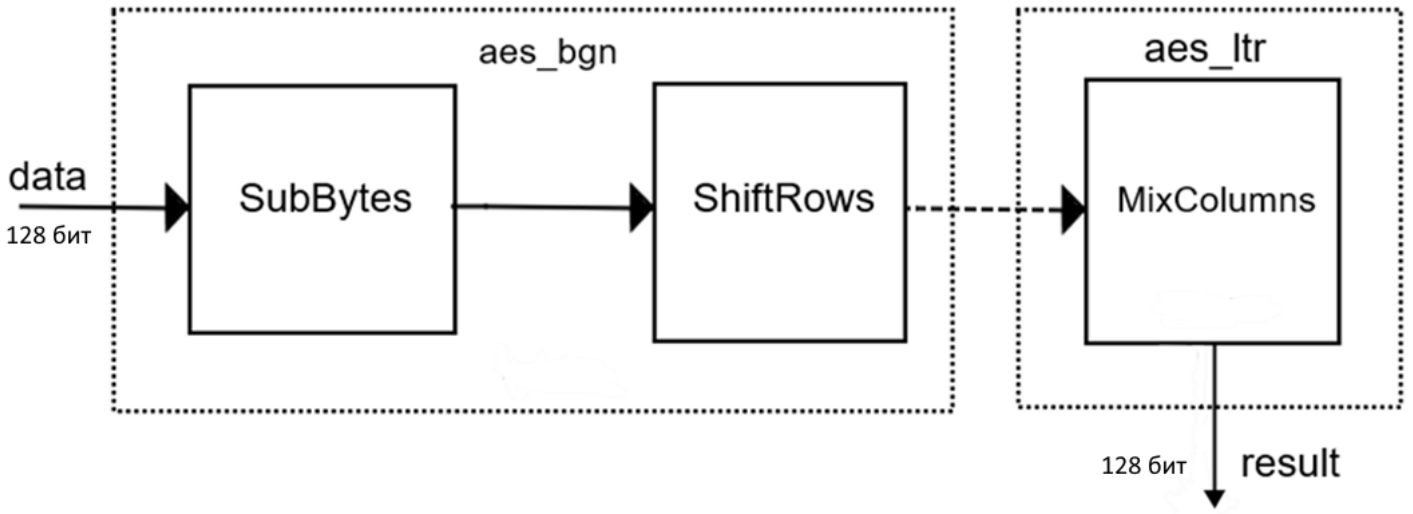


- **В составном поле задача поиска обратного значительно проще**, вследствие чего оптимизация даёт выигрыш по площади в **60%** по сравнению с реализацией табличным поиском

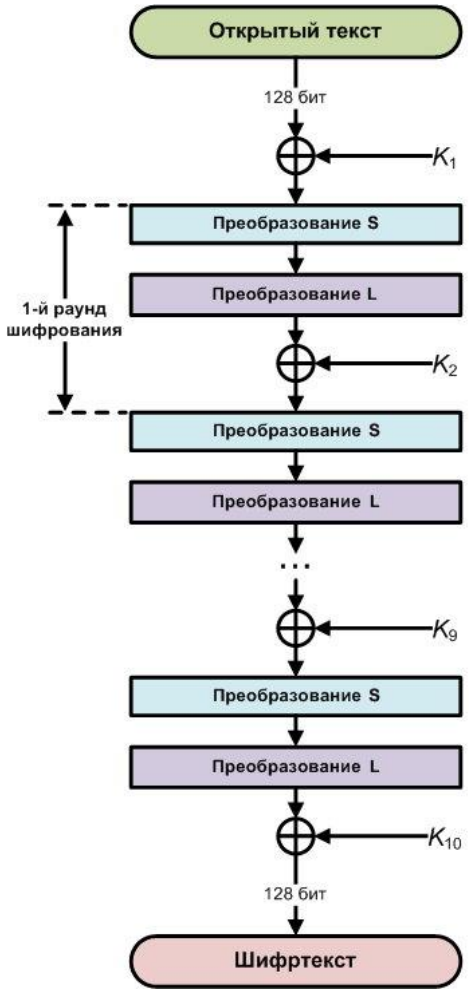
Аппаратная реализация AES

Аппаратная реализация раунда алгоритма AES состоит из двух модулей:

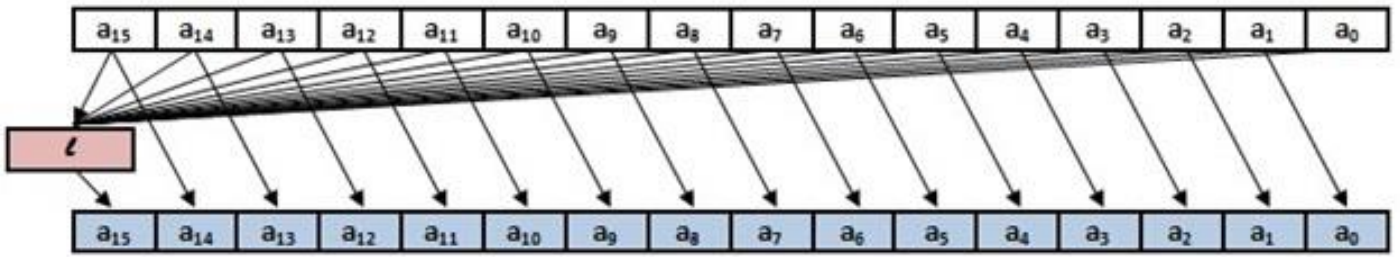
1) Модуль вычисления прямой или обратной табличной замены байт и циклическим байтовым сдвигом
2) Модуль вычисления прямого или обратного линейного преобразования



Кузнечик (ГОСТ Р 34.12 – 2018)



Симметричный шифр с 10 типовыми раундами.
Размер блока – 128 бит.



$$\ell(a_{15}, \ldots, a_0) = \nabla(148 \cdot \Delta(a_{15}) + 32 \cdot \Delta(a_{14}) + 133 \cdot \Delta(a_{13}) + 16 \cdot \Delta(a_{12}) +$$
$$194 \cdot \Delta(a_{11}) + 192 \cdot \Delta(a_{10}) + 1 \cdot \Delta(a_9) + 251 \cdot \Delta(a_8) + 1 \cdot \Delta(a_7) + 192 \cdot \Delta(a_6) +$$
$$194 \cdot \Delta(a_5) + 16 \cdot \Delta(a_4) + 133 \cdot \Delta(a_3) + 32 \cdot \Delta(a_2) + 148 \cdot \Delta(a_1) + 1 \cdot \Delta(a_0))$$

$GF(2^8)$ по модулю неприводимого многочлена $x^8 + x^7 + x^6 + x + 1$.

L-преобразование - оптимизация

l – преобразование:

$$\begin{bmatrix}
94 & 20 & 85 & 10 & C2 & C0 & 01 & FB & 01 & C0 & C2 & 10 & 85 & 20 & 94 & 01 \\
01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 00 & 00 \\
00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 00 & 01 & 00
\end{bmatrix}$$

^16

L – преобразование:

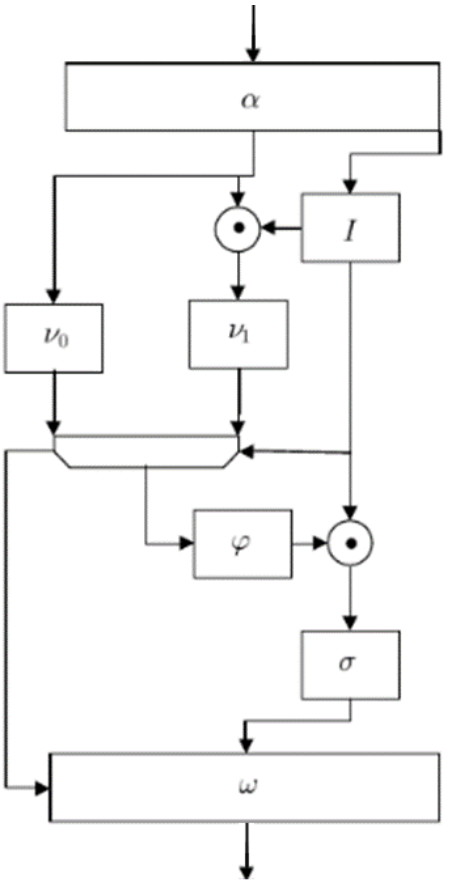
$$\begin{bmatrix}
CF & 98 & 74 & BF & 93 & 8E & F2 & F3 & 0A & BF & F6 & A9 & EA & 8E & 4D & 6E \\
6E & 20 & C6 & DA & 90 & 48 & 89 & 9C & C1 & 64 & B8 & 2D & 86 & 44 & D0 & A2 \\
A2 & C8 & 87 & 70 & 68 & 43 & 1C & 2B & A1 & 63 & 30 & 6B & 9F & 30 & E3 & 76 \\
76 & 33 & 10 & 0C & 1C & 11 & D6 & 6A & A6 & D7 & F6 & 49 & 07 & 14 & E8 & 72 \\
72 & F2 & 6B & CA & 20 & EB & 02 & A4 & 8D & D4 & C4 & 01 & 65 & DD & 4C & 6C \\
6C & 76 & EC & 0C & C5 & BC & AF & 6E & A3 & E1 & 90 & 58 & 0E & 02 & C3 & 48 \\
48 & D5 & 62 & 17 & 06 & 2D & C4 & E7 & D5 & EB & 99 & 78 & 52 & F5 & 16 & 7A \\
7A & E6 & 4E & 1A & BB & 2E & F1 & BE & D4 & AF & 37 & B1 & D4 & 2A & 6E & B8 \\
B8 & 49 & 87 & 14 & CB & 8D & AB & 49 & 09 & 6C & 2A & 01 & 60 & 8E & 4B & 5D \\
5D & D4 & B8 & 2F & 8D & 12 & EE & F6 & 08 & 54 & 0F & F3 & 98 & C8 & 7F & 27 \\
27 & 9F & BE & 68 & 1A & 7C & AD & C9 & 84 & 2F & EB & FE & C6 & 48 & A2 & BD \\
BD & 95 & 5E & 30 & E9 & 60 & BF & 10 & EF & 39 & EC & 91 & 7F & 48 & 89 & 10 \\
10 & E9 & D0 & D9 & F3 & 94 & 3D & AF & 7B & FF & 64 & 91 & 52 & F8 & 0D & DD \\
DD & 99 & 75 & CA & 97 & 44 & 5A & E0 & 30 & A6 & 31 & D3 & DF & 48 & 64 & 84 \\
84 & 2D & 74 & 96 & 5D & 77 & 6F & DE & 54 & B4 & 8D & D1 & 44 & 3C & A5 & 94 \\
94 & 20 & 85 & 10 & C2 & C0 & 01 & FB & 01 & C0 & C2 & 10 & 85 & 20 & 94 & 01
\end{bmatrix}$$

- Каждый байт раскрывается в битовую матрицу 8x8 => получается итоговая матрица 128x128 бит, умножение на которую выполняется в поле GF(2), **аналогично линейному преобразованию AES**

Оптимизация S-преобразования

Декомпозиция табличной замены S-box π:

1. α – умножение на матрицу 8x8 бит
2. I – поиск обратного в поле Галуа $GF(2^4)$
3. ν_0, ν_1, ϕ, σ – табличные замены полубайтовых блоков
4. Старший полубайт будет выбран в зависимости от младшего полубайта:
   a) Младший 0 => старший будет выбран заменой ν_0
   b) Младший отличен от 0 => старший – замена ν_1
5. ω – умножение на матрицу 8x8 бит

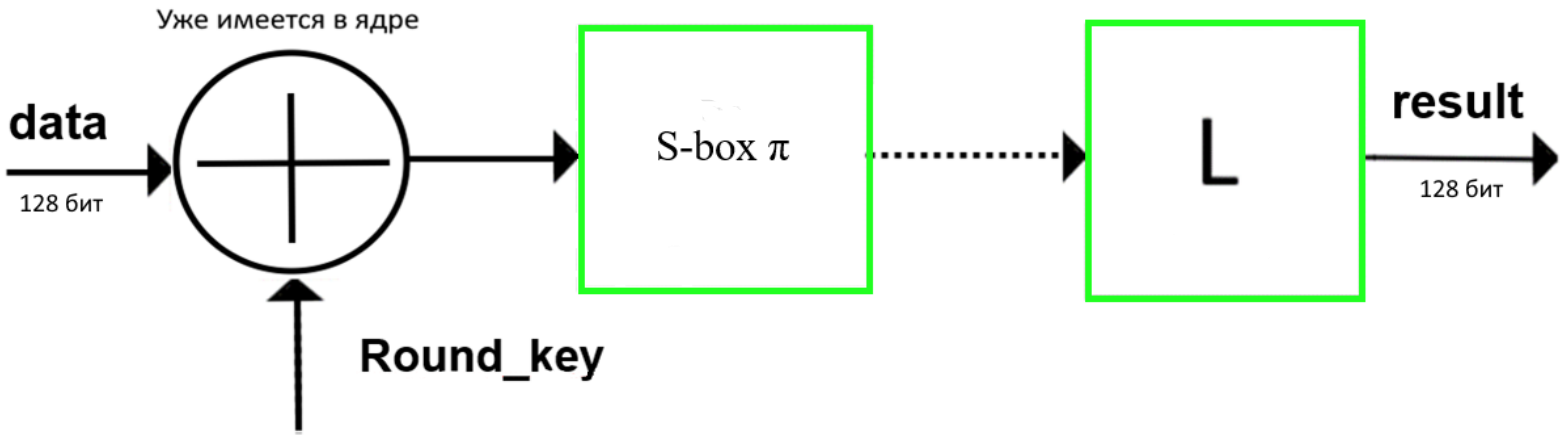


Данная оптимизация даёт выигрыш в 40% по сравнению с реализацией табличным поиском

Аппаратная реализация «Кузнечик»

Аппаратная реализация раунда алгоритма «Кузнечик» состоит из двух модулей:

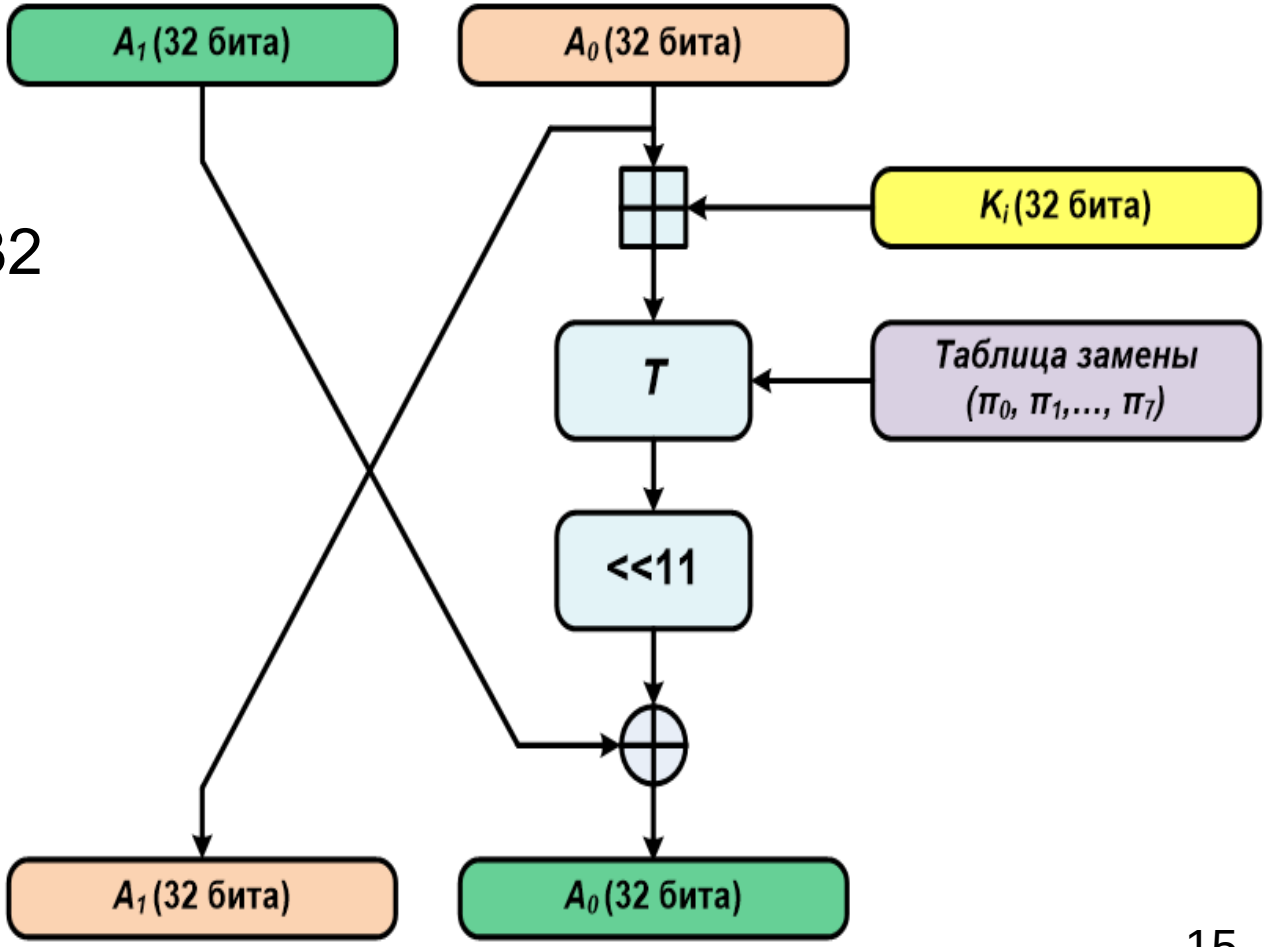
1) Модуль вычисления табличной замены байт.
2) Модуль вычисления линейного преобразования.



Магма (ГОСТ Р 34.12 – 2018)

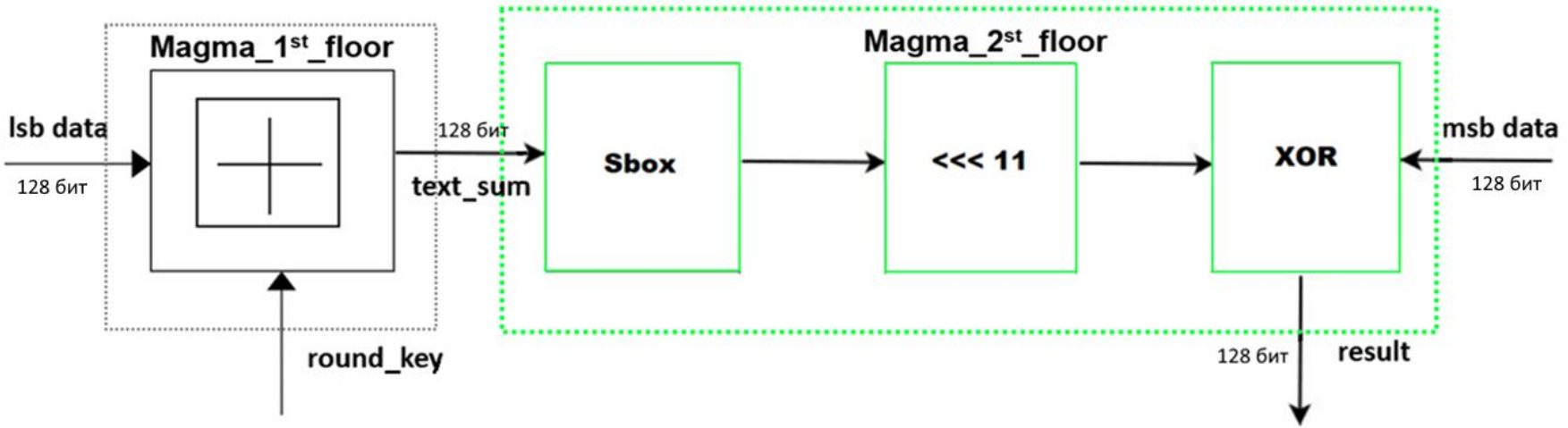
Симметричный шифр с 32 типовыми раундами.

Размер блока – 64 бит.



Аппаратная реализация «Магма»

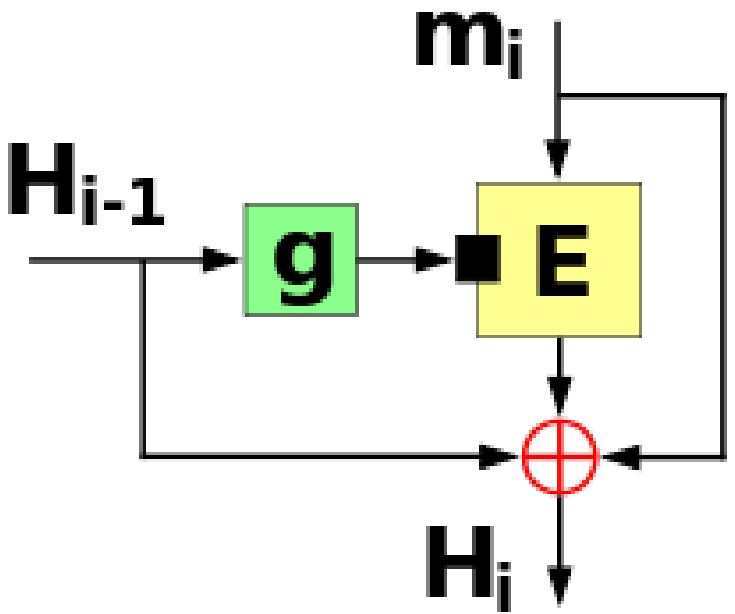
- Аппаратная реализация раунда шифрования Магма состоит из модуля производящего табличную замену, циклический сдвиг и XOR двух половин блока, который способен обрабатывать четыре блока параллельно.



- Для суммирования по модулю с раундовым ключом используется уже реализованный в ядре блок суммирования.

Стрибог (ГОСТ Р 34.11 – 2018)

- Криптографический алгоритм вычисления хеш-функции с размером блока входных данных 512 бит и размером хеш-кода 256 или 512 бит.



- H_{i-1} – предыдущее значение хеш-функции
- H_i – следующее значение хеш-функции
- m_i – блок исходного сообщения
- g – формирование ключа
- E – функция преобразования

Аппаратная реализация «Стрибог»

- Основным шагом алгоритма хэширования Стрибог является связка функций LPS:
  - L – линейное преобразование в виде логического произведения 64 битного блока на матрицу 64 x 64 бит – необходимо реализовать
  - P – транспонирование матрицы байт – есть поддержка в СК
  - S – табличная замена байт по S-box π - та же, что и в алгоритме Кузнечик

- **Была разработана аппаратная реализация функции L**

Отладка

- Для каждого алгоритма была найдена программная реализация и сгенерирован набор эталонных значений соответствующих преобразований
- Была проведена инженерная отладка с использованием сгенерированных эталонов
- Модули встроены в АЛК и проходят системную верификацию
- Все найденные на данный момент ошибки исправлены

Первичный синтез

Значения площади модулей после первичного синтеза в

AES		“Кузнечик”		“Магма”		“Стрибог”	
aes_bgn	3709	kuzn_sbox	2115	2nd_floor	336	kuzn_sbox	2115
aes_ltr	470	kuzn_ltr	2918	-	-	strib_ltr	1401

Влияние оптимизаций на площадь:

Модуль	С оптимизацией	Без оптимизации
aes_bgn	3709	6275
kuzn_sbox	2115	3519

Скорость шифрования

Благодаря тому, что все модули производят вычисления за 1 такт, возможно рассчитать пиковые значения скорости шифрования:

Алгоритм	Время обработки блока (тактов)		
	Эльбрус-v5	Эльбрус-v7	Intel +AVX-512 (SkyLake)
“Кузнечик”	256	10.75	170
“Магма”	234	16.25	29
“AES”	177	10.8	2.56
“Стрибог”	1124	181	432

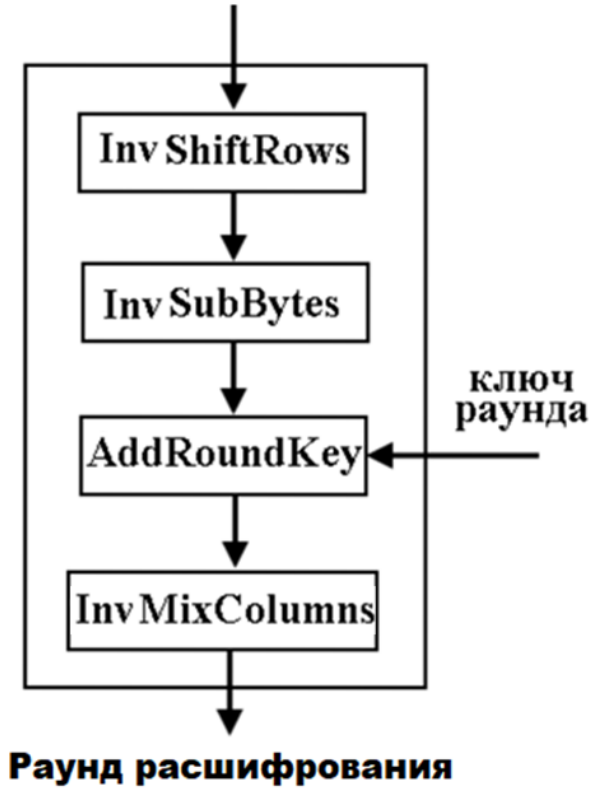
Результаты

По результатам работы были реализованы, оптимизированы и отлажены модули аппаратного ускорения:

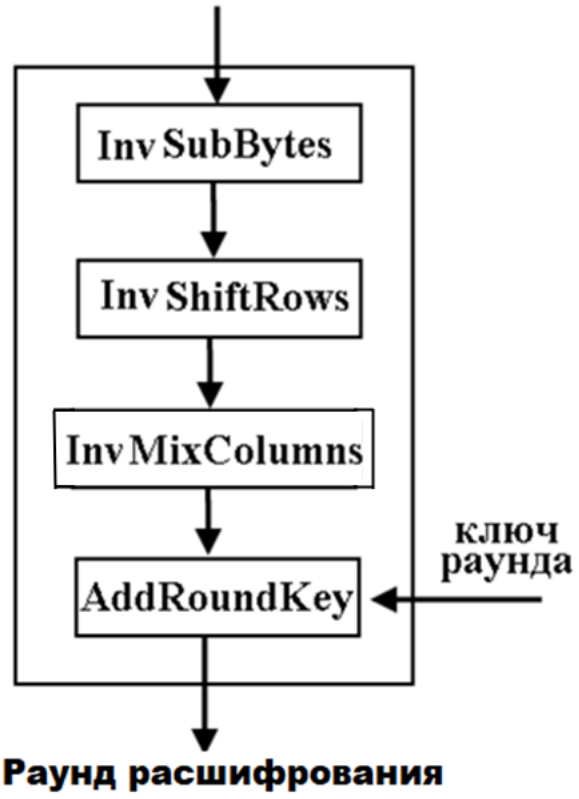
1. Шифрования и расшифрования AES
2. Шифрования “Кузнечик”
3. Шифрования “Магма”
4. Вычисления хеш-функции “Стрибог”

AES (NIST FIPS 197)

Прямой порядок расшифрования:



Измененный порядок расшифрования:



Стрибог (ГОСТ Р 34.11 – 2018)

- $g = LPSX[H_{i-1}](N)$, где $X[k](a) = k \text{ XOR } a$
- $E(K, m) = X[K_{13}]\ LPSX[K_{12}] \ldots LPSX[K_2]\ LPSX[K_1](m)$
- $K_i = LPS(K_{i-1} \text{ XOR } C_{i-1})$, где $K_1 = K$ – изначальный ключ от g