# Алгоритм блочного шифрования BelT

С. В. Агиевич[1], В. А. Галинский[1], Н. Д. Микулич[2], Ю. С. Харин[1]

[1] Национальный научно-исследовательский центр прикладных проблем
математики и информатики Белгосуниверситета
пр. Ф. Скорины, 4, Минск, Беларусь

[2] Государственный центр безопасности информации
при Президенте Республики Беларусь
ул. Кальварийская, 7а, Минск, Беларусь

## 1 Введение

К настоящему времени разработано большое количество алгоритмов блочного шифрования, многие из которых являются национальными или ведомственными стандартами. Наибольшую известность приобрели криптоалгоритмы `DES`, `ГОСТ 28147-89`, `IDEA`, `Rijndael` и др. Они находятся под пристальным вниманием криптологов, изучающих слабости и оценивающих стойкость этих криптосистем.

В настоящей статье описывается новый алгоритм блочного шифрования с длиной блока 128 битов и 256-битовым ключом. Приводятся принципы проектирования алгоритма и характеристики быстродействия.

## 2 Спецификация алгоритма

### 2.1 Обозначения

| | |
|---|---|
| $\mathcal{A}^n$ | множество всех слов длины $n$ в алфавите $\mathcal{A} = \{0, 1\}$; |
| $u \parallel v$ | конкатенация $u_1u_2 \dots u_nv_1v_2 \dots v_m$ слов $u = u_1u_2 \dots u_n$ и $v = v_1v_2 \dots v_m$; |
| $\bar{u}$ | а) число $2^7u_1 + 2^6u_2 + \dots + u_8$ для слова $u = u_1u_2 \dots u_8 \in \mathcal{A}^8$ и<br>б) число $\bar{u}_1 + 2^8\bar{u}_2 + 2^{16}\bar{u}_3 + 2^{24}\bar{u}_4$ для слова $u = u_1 \parallel u_2 \parallel u_3 \parallel u_4$, $u_i \in \mathcal{A}^8$; |
| $\langle U \rangle_{32}$ | слово $u \in \mathcal{A}^{32}$ такое, что $\bar{u} \equiv U \pmod{2^{32}}$; |
| $u \oplus v$ | поразрядная по модулю 2 сумма двоичных слов $u, v \in \mathcal{A}^{32}$; |
| $u \boxplus v$ | слово $\langle \bar{u} + \bar{v} \rangle_{32}$ для слов $u, v \in \mathcal{A}^{32}$; |
| $u \boxminus v$ | слово $w \in \mathcal{A}^{32}$ такое, что $u = v \boxplus w$; |
| $\lambda(u)$ | слово $\left\langle 2\bar{u} + \left\lfloor \frac{\bar{u}}{2^{31}} \right\rfloor \right\rangle_{32}$ для слова $u \in \mathcal{A}^{32}$; |
| $\lambda^r(u)$ | слово, полученное $r$-кратным действием $\lambda$ на $u$; |
| $a \leftarrow u$ | запись значения $u$ в регистр $a$; |
| $a \leftrightarrow b$ | перестановка значений регистров $a$ и $b$. |

### 2.2 Алгоритм

**Данные.** Алгоритм предназначен для криптографического преобразования данных — зашифрования и расшифрования слов $X \in \mathcal{A}^{128}$ на ключе $\theta \in \mathcal{A}^{256}$.

**Расписание ключей.** Ключу $\theta = \theta_1 \parallel \theta_2 \parallel \dots \parallel \theta_8$, $\theta_i \in \mathcal{A}^{32}$, ставятся в соответствие слова $\kappa_1, \kappa_2, \dots, \kappa_{56} \in \mathcal{A}^{32}$. При зашифровании $\kappa_i$ является $i$-м элементом последовательности

$$\theta_1, \theta_2, \dots, \theta_8, \theta_1, \theta_2, \dots, \theta_8, \dots$$

При расшифровании $\kappa_i$ является $i$-м элементом последовательности

$$\theta_8, \theta_7, \ldots, \theta_1, \theta_8, \theta_7, \ldots, \theta_1, \ldots$$

**Структура алгоритма.** Для криптопреобразования слова $X_1 \parallel X_2 \parallel X_3 \parallel X_4$, $X_i \in \mathcal{A}^{32}$, используются 32-разрядные регистры $a, b, c, d$, первоначально содержащие значения

$$a \leftarrow X_1, \quad b \leftarrow X_2, \quad c \leftarrow X_3, \quad d \leftarrow X_4.$$

Криптопреобразование состоит в выполнении объединенных в такты вычислений над содержимым регистров. Дополнительно используется вспомогательный 32-разрядный регистр $e$.

При зашифровании выполняются такты $1, 2, \ldots, 8$. Результатом зашифрования является слово $b \parallel d \parallel a \parallel c$.

При расшифровании выполняются такты $8, 7, \ldots, 1$. Результатом расшифрования является слово $c \parallel a \parallel d \parallel b$.

**Такт.** На $t$-м такте зашифрования выполняются шаги (см. рис. 1):

$$\begin{array}{ll}
(1) & b \leftarrow b \oplus G_5(a \boxplus \kappa_{7t-6}), \\
(2) & c \leftarrow c \oplus G_{21}(d \boxplus \kappa_{7t-5}), \\
 & \\
(3) & a \leftarrow a \boxminus G_{13}(b \boxplus \kappa_{7t-4}), \\
 & \\
(4) & e \leftarrow G_{21}(b \boxplus c \boxplus \kappa_{7t-3}) \oplus \langle t \rangle_{32}, \\
(5) & b \leftarrow b \boxplus e, \\
(6) & c \leftarrow c \boxminus e, \\
 & \\
(7) & d \leftarrow d \boxplus G_{13}(c \boxplus \kappa_{7t-2}), \\
 & \\
(8) & b \leftarrow b \oplus G_{21}(a \boxplus \kappa_{7t-1}), \\
(9) & c \leftarrow c \oplus G_5(d \boxplus \kappa_{7t}), \\
 & \\
(10) & a \leftrightarrow b, \\
(11) & c \leftrightarrow d, \\
(12) & b \leftrightarrow c.
\end{array}$$

На $t$-м такте расшифрования вместо шага (12) выполняется шаг

$$(12') \quad a \leftrightarrow d.$$

**Преобразование $G_r$.** Преобразование $G_r \colon \mathcal{A}^{32} \to \mathcal{A}^{32}$ ставит в соответствие слову $u = u_1 \parallel u_2 \parallel u_3 \parallel u_4$, $u_i \in \mathcal{A}^8$, слово

$$G_r(u) = \lambda^r \left( S(u_1) \parallel S(u_2) \parallel S(u_3) \parallel S(u_4) \right).$$

**Подстановка $S$.** Подстановка $S \colon \mathcal{A}^8 \to \mathcal{A}^8$ задается таблицей 1. Двоичные слова записываются в шестнадцатеричной системе счисления. При этом последовательным четырем двоичным символам соответствует одна шестнадцатеричная цифра. Например, $01010010 = \mathrm{A2}_{16}$. Если $u = \mathrm{IJ}_{16}$, то значение $S(u)$ находится на пересечении строки I и столбца J. Например, $S(\mathrm{A2}_{16}) = \mathrm{9B}_{16}$.

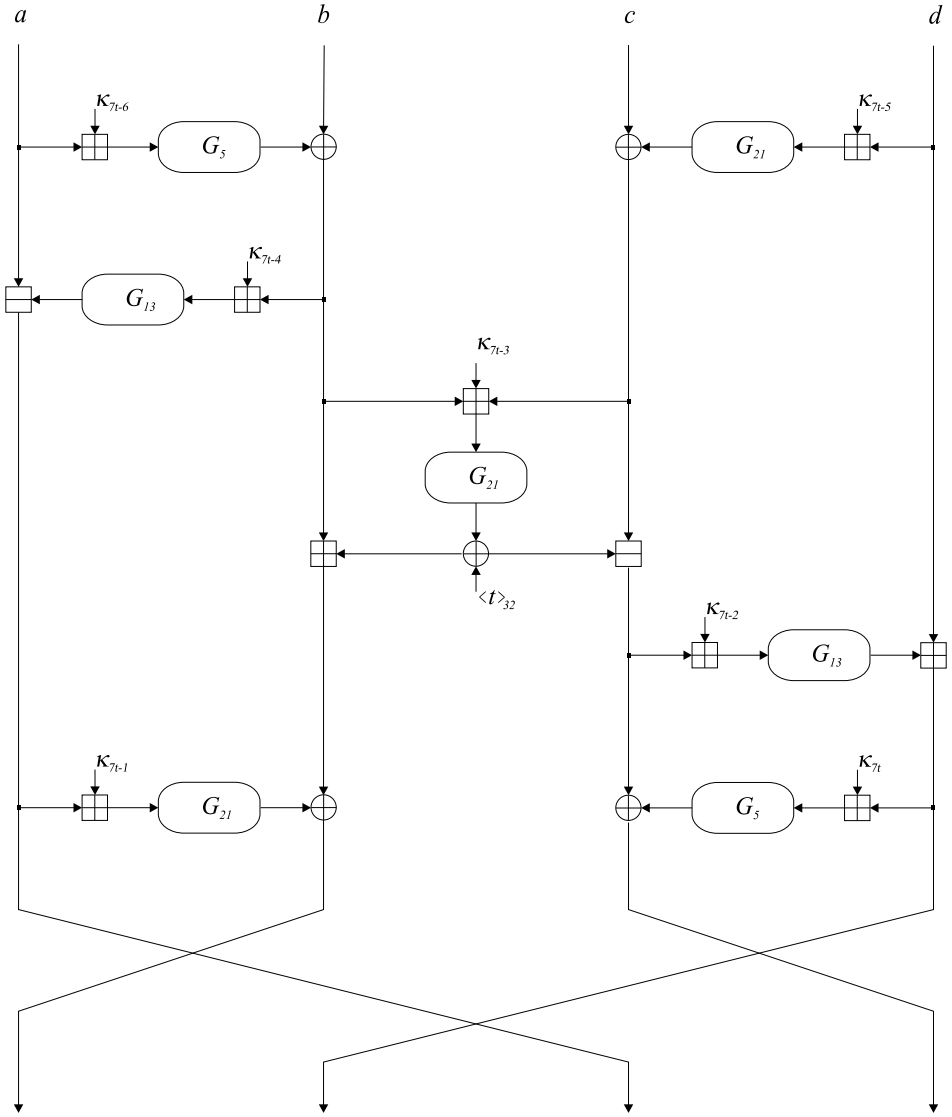


Рис. 1: Вычисления на $t$-м такте зашифрования

## 3 Принципы проектирования алгоритма

Перечислим основные принципы, которыми мы руководствовались при разработке алгоритма.

1. Структурная близость преобразований зашифрования и расшифрования.

   Такт зашифрования BelT структурно незначительно отличается от такта зашифрования. Структурная близость упрощает программную и аппаратную реализации алгоритма.

2. Использование подстановки $S$.

   Возможны ситуации в которых пользователю требуется «личная» криптосистема. Один из способов персонализации алгоритма состоит в использовании подстановки $S$ в качестве дополнительного долговременного ключа.

3. Конструкция и характеристики подстановки $S$.

   Таблицы истинности координатных булевых функций подстановки $S$ выбирались как различные отрезки длины 255 линейных рекуррентных последовательностей (см. [1])

Таблица 1: Подстановка $S$

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | B1 | 94 | BA | C8 | 0A | 08 | F5 | 3B | 36 | 6D | 00 | 8E | 58 | 4A | 5D | E4 |
| 1 | 85 | 04 | FA | 9D | 1B | B6 | C7 | AC | 25 | 2E | 72 | C2 | 02 | FD | CE | 0D |
| 2 | 5B | E3 | D6 | 12 | 17 | B9 | 61 | 81 | FE | 67 | 86 | AD | 71 | 6B | 89 | 0B |
| 3 | 5C | B0 | C0 | FF | 33 | C3 | 56 | B8 | 35 | C4 | 05 | AE | D8 | E0 | 7F | 99 |
| 4 | E1 | 2B | DC | 1A | E2 | 82 | 57 | EC | 70 | 3F | CC | F0 | 95 | EE | 8D | F1 |
| 5 | C1 | AB | 76 | 38 | 9F | E6 | 78 | CA | F7 | C6 | F8 | 60 | D5 | BB | 9C | 4F |
| 6 | F3 | 3C | 65 | 7B | 63 | 7C | 30 | 6A | DD | 4E | A7 | 79 | 9E | B2 | 3D | 31 |
| 7 | 3E | 98 | B5 | 6E | 27 | D3 | BC | CF | 59 | 1E | 18 | 1F | 4C | 5A | B7 | 93 |
| 8 | E9 | DE | E7 | 2C | 8F | 0C | 0F | A6 | 2D | DB | 49 | F4 | 6F | 73 | 96 | 47 |
| 9 | 06 | 07 | 53 | 16 | ED | 24 | 7A | 37 | 39 | CB | A3 | 83 | 03 | A9 | 8B | F6 |
| A | 92 | BD | 9B | 1C | E5 | D1 | 41 | 01 | 54 | 45 | FB | C9 | 5E | 4D | 0E | F2 |
| B | 68 | 20 | 80 | AA | 22 | 7D | 64 | 2F | 26 | 87 | F9 | 34 | 90 | 40 | 55 | 11 |
| C | BE | 32 | 97 | 13 | 43 | FC | 9A | 48 | A0 | 2A | 88 | 5F | 19 | 4B | 09 | A1 |
| D | 7E | CD | A4 | D0 | 15 | 44 | AF | 8C | A5 | 84 | 50 | BF | 66 | D2 | E8 | 8A |
| E | A2 | D7 | 46 | 52 | 42 | A8 | DF | B3 | 69 | 74 | C5 | 51 | EB | 23 | 29 | 21 |
| F | D4 | EF | D9 | B4 | 3A | 62 | 28 | 75 | 91 | 14 | 10 | EA | 77 | 6C | DA | 1D |

с примитивным характеристическим многочленом

$$p(\lambda) = \lambda^8 + \lambda^6 + \lambda^5 + \lambda^2 + 1,$$

причем в фиксированную позицию каждого отрезка вставлялся символ 0.

Приведем некоторые криптографические характеристики построенной подстановки:

1) нелинейность $S$ равняется 102 (при выбранных размерностях нелинейность $S$ не может превышать 120, высокая нелинейность обеспечивает стойкость криптосистемы к методам линейного криптоанализа [7]);
2) разностные характеристики $S$: $R_{\oplus\oplus}(S) = 8$, $R_{\boxplus\boxplus}(S) = 7$, $R_{\oplus\boxplus}(S) = 6$, $R_{\boxplus\oplus}(S) = 3$, где, например,

$$R_{\boxplus\oplus}(S) = \max_{\substack{\alpha,\beta\in\mathcal{A}^8 \\ \alpha\neq 0}} \sum_{x\in\mathcal{A}^8} \mathrm{I}\{S(x \boxplus \alpha) = S(x) \oplus \beta\},$$

   $\mathrm{I}\{\mathcal{E}\}$ — индикатор наступления события $\mathcal{E}$ (малые значения разностных характеристик затрудняют применение методов разностного анализа [3]);
3) степени всех координатных булевых функций отображения $S$ равняются 7 (при выбранных размерностях это максимально возможное значение, высокие степени координатных функций затрудняют применение некоторых модификаций разностных атак [6]);
4) отсутствуют квадратичные соотношения, связывающие входы и выходы $S$-блока (наличие таких соотношений упрощает применение алгебраических атак [4]).

4. Чередование операций.

   При выполнении тактов алгоритма к содержимому регистров добавляются (по правилам $\oplus$, $\boxplus$, $\boxminus$) образы $G$-преобразования. При этом операция покомпонентного суммирования и аддитивные операции по модулю $2^{32}$ чередуются. Например, при зашифровании к содержимому регистра $b$ добавляется

   ($\oplus$) значение $G_5(a \boxplus \kappa_1)$,

(⊞) значение $G_{21}(b \boxplus c \boxplus \kappa_4) \oplus \langle 1 \rangle_{32}$,

(⊕) значение $G_{21}(a \boxplus \kappa_6)$,

затем содержимое $b$ переписывается в регистр $a$, из него вычитается

(⊟) значение $G_{13}(b \boxplus \kappa_{10})$

и т. д. Использование операций из разных групп и их чередование затрудняет применение некоторых методов криптоанализа.

5. Неоднородные тактовые преобразования.

   На шаге (4) тактовых преобразований содержимое регистра $e$ суммируется с номером такта $\langle t \rangle_{32}$ и все такты являются различными подстановками даже при одинаковых наборах тактовых параметров $k_t = (\kappa_{7t-6}, \ldots, \kappa_{7t})$, $t = 1, \ldots, 8$. При этом снижается вероятность появления у анализируемых подстановок зашифрования $F_\theta$, $\theta \in \mathcal{A}^{256}$, алгебраических слабостей таких, например, как

   – низкий порядок подстановок $F_\theta$ и, в частности, наличие слабых ключей, т. е. таких $\theta$, что $F_\theta^2$ — тождественная подстановка;
   – наличие эквивалентных ключей, т. е. таких различных $\theta_1, \theta_2$, что $F_{\theta_1} = F_{\theta_2}$.

6. Выбор числа тактов.

   Криптопреобразования алгоритма являются композициями 8 тактовых подстановок. При разработке алгоритма был проведен первичный статистический анализ, направленный на обнаружение статистических зависимостей между открытым текстом, шифртекстом и ключом. Установлено, что уже для 2-тактовых криптопреобразований значимые статистические закономерности отсутствуют.

## 4 Быстродействие

При разработке алгоритма в качестве базовой была выбрана 32-разрядная архитектура. Единственная не 32-разрядная операция — это замена байта на блоке $S$. При программной реализации алгоритма нами использовалась следующая оптимизация:

(а) Для $r = 5, 13, 21, 29$ предварительно строятся расширенные таблицы, задающие отображения

$$S_r \colon \mathcal{A}^8 \to \mathcal{A}^{32}, \quad x \mapsto \lambda^r(S(x) \parallel 0^{24}),$$

где $0^{24}$ — нулевое слово длины 24. Для хранения всех четырех таблиц требуется 4 Кб памяти.

(б) Значение $G_5(x)$, $x = x_1 \parallel x_2 \parallel x_3 \parallel x_4$, $x_i \in \mathcal{A}^8$, вычисляется как

$$G_5(x) = S_5(x_1) \oplus S_{13}(x_2) \oplus S_{21}(x_3) \oplus S_{29}(x_4).$$

Аналогично:

$$G_{13}(x) = S_{13}(x_1) \oplus S_{21}(x_2) \oplus S_{29}(x_3) \oplus S_5(x_4),$$
$$G_{21}(x) = S_{21}(x_1) \oplus S_{29}(x_2) \oplus S_5(x_3) \oplus S_{13}(x_4).$$

Количество операций, требуемых для зашифрования (расшифрования) 16-байтового блока данных, приводится в таблице 2.

Таблица 2: Количество операций для криптопреобразования 16-байтового блока

| Операции | Количество |
|---|---|
| $\boxplus$, $\boxminus$ | 96 |
| $\oplus$ | 40 |
| $\lambda^r$ | 56 |
| $S$ | 224 |
| $\leftrightarrow$ | 23 |
| Общее количество | 439 |

Производительность алгоритма сравнивалась с производительностью криптосистем конкурса AES — кандидатов на стандарт шифрования США [2]. Нами были отобраны криптосистемы `RC6`, `Rijndael`, `MARS`, `Twofish`, `Serpent`, `DFC`, `Safer+`. Первые пять из них прошли во второй тур AES и обладают лучшими характеристиками быстродействия среди всех 15 кандидатов. Криптосистема `Rijndael` победила в конкурсе AES.

Для сравнения использовались программные реализации криптосистем, написанные Б. Гладманом [5] на языке C. Учитывались соглашения данных реализаций и использовались только те способы оптимизации вычислений, которые имеются в данных реализациях. Например, описанный выше способ вычисления значения $G_r$, основанный на использовании расширенных таблиц подстановки, применялся в программной реализации криптосистемы `Rijndael`, криптосистема `Serpent` по неофициальным данным была второй.

Сравнение быстродействия осуществлялось с помощью утилиты VTune Perfomance Analyzer [8]. Данная утилита позволяет определять число тактов, необходимых для выполнения программы на процессорах семейства Pentium и учитывает специфику аппаратного строения процессоров (использование конвейерной архитектуры, кэша команд, кэша данных и др.). Результаты сравнительного анализа приведены в таблице 3. Из таблицы, например, следует, что предлагаемый алгоритм позволяет выполнять шифрование на процессоре Pentium-200 со скоростью $200 \cdot \frac{128}{1367} \approx 18.73$ Мбит/сек., а на процессоре Pentium III-1000 — со скоростью $1000 \cdot \frac{128}{682} \approx 187.7$ Мбит/сек.

Таблица 3: Числов тактов процессоров семейства Pentium для зашифрования 16 байтов открытого текста

| Процессор | Кандидаты AES | | | | | | | BelT |
|---|---|---|---|---|---|---|---|---|
| | `RC6` | `Rijndael` | `MARS` | `Twofish` | `Serpent` | `DFC` | `Safer+` | |
| `Pentium` | 920 | 1022 | 954 | 892 | 1693 | 3750 | 3789 | 1367 |
| `Pentium II` | 271 | 399 | 388 | 404 | 964 | 1246 | 1910 | 642 |
| `Pentium III` | 281 | 427 | 410 | 429 | 984 | 1256 | 1928 | 682 |

Дополнительным достоинством алгоритма является быстрый, фактически не требующий вычислений, способ определения тактовых ключей $\kappa_1, \ldots, \kappa_{56}$. Сложное расписание снижает эффективность применения криптосистемы в тех случаях, когда требуется частая смена ключей, например, в некоторых схемах построения хэш-функции. Для сравнения, вычислительная сложность построения расписания ключей в алгоритме `Twofish` примерно в 5 раз превышает сложность криптопреобразования 16-байтового блока.

## Список литературы


1. *Лидл Р.*, *Нидеррайтер Г.* Конечные поля: В 2 т. — М.: Мир, 1988.

2. Advanced Encryption Standard (AES) development effort. — Avail. at: `http://csrc.nist.gov/encryption/aes/index2.htm`, 2001.

3. *Biham E.*, *Shamir A.* Differential cryptanalysis of DES-like cryptosystems. — J. Cryptology, 1991, v. 4, p. 3–72.

4. *Courtois N. T.*, *Pieprzyk J.* Cryptanalysis of block ciphers with overdefined systems of equations. — IACR Eprint Server, avail. at: `http://eprint.iacr.org/2002/044/`, 2002.

5. *Gladman B.* Implementation experience with AES candidate algorithms. — In: Proceedings: Second AES Candidate Conference (AES2), 1999, avail. at: `http://csrc.nist.gov/CryptoToolkit/aes/round1/conf2/aes2conf.htm`.

6. *Knudsen L. R.* Truncated and higher order differentials. — Lect. Notes Comp. Sci., 1995, v. 1008, p. 196–211.

7. *Matsui M.* Linear Cryptanalysis Method for DES Cipher. — Lect. Notes Comp. Sci., 1994, v. 765, p. 386–397.

8. VTune Perfomance Analyzers. — Avail. at: `http://developer.intel.com/design/perftool/vtune`.