# Лабораторная работа №1: Реализация алгоритма блочного шифрования «Магма» в режиме MGM (ГОСТ 34.13–2018)

# Общее написание алгоритма

Алгоритм «Магма» (ранее ГОСТ 28147-89, ныне ГОСТ Р 34.12–2015) — это симметричный блочный шифр с фиксированными параметрами:

* **Длина блока:** 64 бита (8 байт)
* **Длина ключа:** 256 бит (32 байта)
* **Количество раундов:** 32

Каждый раунд шифрования включает:

* сложение 32-битного полу-блока с раундовым ключом по модулю ,
* нелинейную замену (8 S-блоков),
* циклический сдвиг на 11 бит влево,
* операцию XOR с другим полу-блоком.

#### ****Режим работы MGM (Магма-Galois Mode)****

**MGM** (Mode of Galois Multiplication) — режим работы блочного шифра по ГОСТ 34.13–2018, аналогичный режиму GCM (из международных стандартов). Он обеспечивает одновременно **конфиденциальность и целостность данных** за счёт встроенной имитовставки.

* **Принцип работы:**
  + Используется счётчик (counter) для генерации ключевого потока, как в CTR.
  + Каждый блок данных шифруется путём XOR с гаммой, полученной из шифрования счётчика.
  + Дополнительно используется поле Галуа GF(2⁶⁴) для вычисления имитовставки (MAC), что обеспечивает проверку целостности.
* **Преимущества:**
  + Высокая скорость и параллелизм, так как шифрование блоков не зависит от предыдущих.
  + Поддержка имитовставки (аутентификация).
  + Универсальность — может использоваться и как режим шифрования, и как режим MAC.
* **Недостатки:**
  + Чувствительность к повторному использованию счётчиков/IV: при совпадении IV с одинаковым ключом возможна компрометация данных.
  + Более сложная реализация по сравнению с ECB или OFB.
* **Применение:**
  + Шифрование передаваемых данных в открытых сетях, где важны и конфиденциальность, и защита от подделки (например, защищённые каналы связи, VPN, облачные хранилища).
  + Используется в отечественных криптопровайдерах, соответствующих требованиям ФСТЭК и ФСБ.

# 2. Описание криптографических свойств алгоритма и принципов его построения

1. **Стойкость к атакам:**

* **Полный перебор (brute-force):**  
  Алгоритм использует 256-битный ключ, что обеспечивает теоретическую стойкость на уровне 22562^{256}2256 операций. Это соответствует требованиям современных стандартов и делает перебор невозможным на практике.
* **Линейный и дифференциальный криптоанализ:**  
  Благодаря использованию восьми независимых S-блоков и 32 раундов, алгоритм демонстрирует высокую стойкость к этим видам атак. Вероятность успешной линейной аппроксимации крайне низка.
* **Атаки на сокращённое число раундов:**  
  Даже при снижении числа раундов до 16, не обнаружено эффективных атак. При полном раундовом расписании (32) «Магма» считается криптоустойчивым.

1. **Принципы построения:**

* **Сеть Фейстеля:**  
  Структура шифра основана на классической сбалансированной сети Фейстеля. В каждом раунде правая и левая половины блока обрабатываются попеременно, что обеспечивает обратимость преобразования без необходимости в инверсии.
* **Раундовая функция F:**
  + Сложение с раундовым ключом по модулю 232;
  + Нелинейная замена через S-блоки (8 замен по 4 бита);
  + Циклический сдвиг результата на 11 бит влево;
* **Ключевое расписание:**
  + Ключ длиной 256 бит разбивается на 8 подключей по 32 бита;
  + Для 32 раундов подключи применяются в следующем порядке:

K1..K8, K1..K8, K1..K8, K8..K1,

* Такая схема обеспечивает равномерную диффузию и равномерное распределение подключей по всем раундам.

#### ****Максимальная теоретическая стойкость:****

* **К полному перебору:**

Сложность=O(2256)

Это значение превышает требования стандартов NIST (минимум 2112) и остаётся устойчивым даже в контексте возможных квантовых атак.

* **К дифференциальному анализу:**  
  Исследования показывают, что даже при ослабленном числе раундов (16–24), эффективность дифференциального анализа остаётся крайне низкой. Для полного алгоритма требуется не менее 2128 операций, что делает атаку практически невозможной.

#### ****Известные оценки стойкости (по открытым источникам)****

* **ГОСТ 34.12-2015**: Стандарт не имеет известных практических уязвимостей при корректной реализации.
* **Исследования ТК-26**: Подтверждают, что алгоритм устойчив к:
* Линейному криптоанализу (вероятность линейных характеристик < 2-64).
* Атакам на связанных ключах.
* **Ограничения**:
* Режим OFB требует уникальности IV для каждого сеанса шифрования, иначе ключевой поток повторяется.

#### ****Особенности и ограничения режима MGM****

* **Применение умножения в поле Галуа GF(264):**  
  Используется для вычисления имитовставки (MAC), что обеспечивает устойчивость к модификации зашифрованных данных.
* **Устойчивость режима MGM:**
  + Режим обеспечивает **аутентифицированное шифрование**, то есть одновременно защищает данные от раскрытия и подделки.
  + При корректной реализации обеспечивает защиту от атак типа "chosen ciphertext" (CCA).
* **Ограничения и уязвимости:**
  + Повторное использование одного и того же nonce/IV с одним и тем же ключом приводит к утечке информации — недопустимо.
  + Требуется синхронное согласование начального вектора (IV) между шифрующими и расшифровывающими сторонами.

# 3. Вычислительная сложность алгоритма

#### ****Временная сложность****

**Один раунд шифрования Магма:**

* Сложение с подключом по модулю 232 — **O(1)**
* Нелинейная замена через S-блоки (8 подстановок по 4 бита) — **O(1)**
* Циклический сдвиг на 11 бит — **O(1)**
* XOR с другой половиной блока — **O(1)**
* **Итого на один раунд:** O(1)

**2. Полное шифрование одного блока (64 бита):**

* 32 раунда × O(1) = **O(1)** (всегда фиксированное число операций)

**3. Режим MGM при обработке данных объёмом N байт:**

* Число блоков: N/8
* Для каждого блока:
  + Генерация гаммы (шифрование счётчика): O(1)
  + XOR с открытым текстом: O(1)
  + MAC-обновление через умножение в GF(2⁶⁴): O(1)
* **Итоговая сложность:** **O(N)** — линейная по объёму входных данных

**4. Хеширование/имитовставка в MGM:**

* Каждому блоку сопоставляется MAC-компонент с умножением в поле Галуа.
* Эти операции выполняются за **постоянное время** на каждом шаге (в силу фиксированной длины блока) → **O(N)**

#### ****Потребление памяти****

**1. Постоянная память (статическая):**

* **S-блоки:** 8 таблиц по 16 байт = **128 байт**
* **Раундовые ключи:** 32 × 4 байта = **128 байт**
* **Контекстные структуры режима MGM:**
  + Счётчик (CTR): 8 байт
  + Гамма-блок: 8 байт
  + MAC-блок: 8 байт

**2. Динамическая память (при шифровании N байт):**

* Входной буфер: O(N)
* Выходной буфер: O(N)
* Промежуточные буферы: незначительны (8–16 байт)

**3. Стек вызовов:**

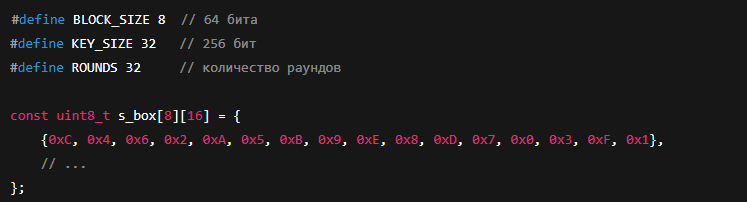
* Алгоритм реализован без рекурсии
* Все функции имеют предсказуемую и фиксированную глубину вызова
* Стековая нагрузка — минимальна

#### ****Оптимизации и узкие места (bottlenecks)****

* **Ключевое расписание:**  
  Выполняется один раз за сеанс шифрования и имеет сложность O(1).  
  Однако при частой смене ключа (например, каждые 10 блоков — как в стресс-тестах) ключевое расписание становится узким местом и увеличивает общее время работы.
* **Побитовые и табличные операции:**
  + Все операции реализованы с учётом быстродействия современных CPU (сложение, XOR, сдвиги, таблицы S-блоков).
  + Не используются медленные операции деления или ветвлений.
* **Параллелизация:**
  + Режим MGM допускает параллельную обработку блоков данных при шифровании (так как счётчики независимы).
  + Однако в данной реализации используется последовательная обработка (однопоточная).

# 4. Общее описание программной реализации алгоритма

## 1. Основные константы и S-блоки

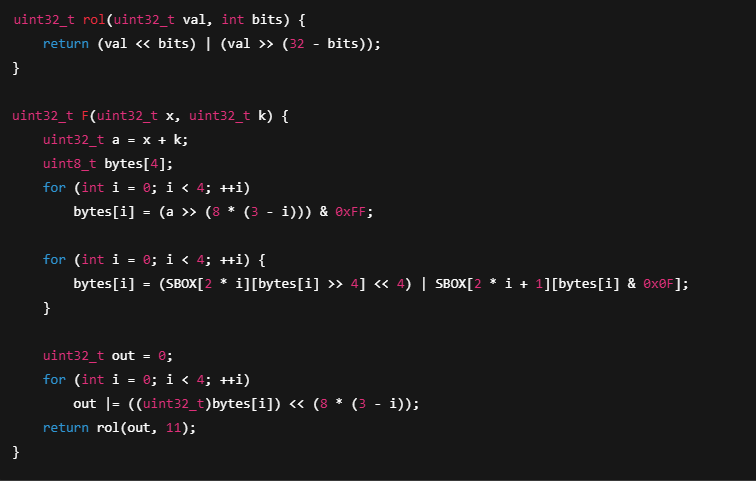


**Описание**:  
В этом разделе определены ключевые параметры алгоритма:

* BLOCK\_SIZE - размер блока данных (64 бита)
* KEY\_SIZE - размер ключа шифрования (256 бит)
* ROUNDS - количество раундов шифрования (32)
* s\_box - набор S-блоков согласно ГОСТ 34.12-2015

**Значимость**:  
Эти константы формируют основу для всех последующих криптографических преобразований. S-блоки обеспечивают нелинейность преобразований, что критически важно для стойкости алгоритма.

**2. Базовые криптографические операции**

**

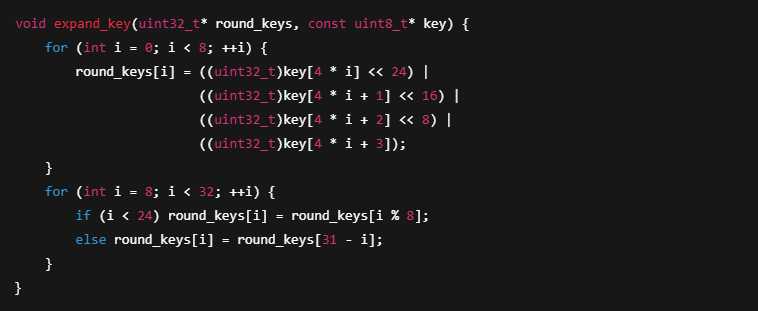
**Описание:**

* rol — выполняет циклический сдвиг 32-битного слова на заданное число бит.
* F — основная нелинейная функция раунда:
  + сложение с подключом,
  + подстановка через S-блоки,
  + сдвиг на 11 бит.

**Значимость:**

* F() используется в каждом раунде и обеспечивает путаницу и рассеивание.
* Оптимизирована по скорости: все операции фиксированы и побитовые.

**3. Генерация раундовых ключей**

**

**Описание:**

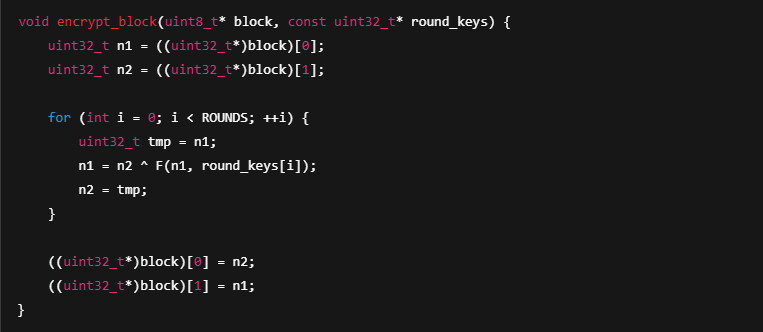
* Ключ длиной 256 бит разбивается на 8×32-битных подключей.
* 24 ключа получаются повторением, а последние 8 — в обратном порядке.

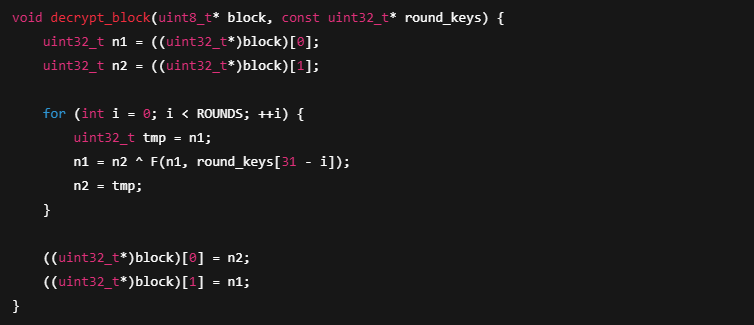
**Схема:**  
K1..K8, K1..K8, K1..K8, K8..K1

**Значимость:**

* Простота схемы компенсируется числом раундов.
* Гарантирует равномерное распределение подключей по раундам.

**4. Шифрование и расшифрование блока**





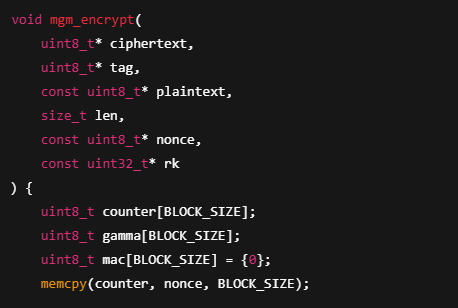
**Логика работы:**

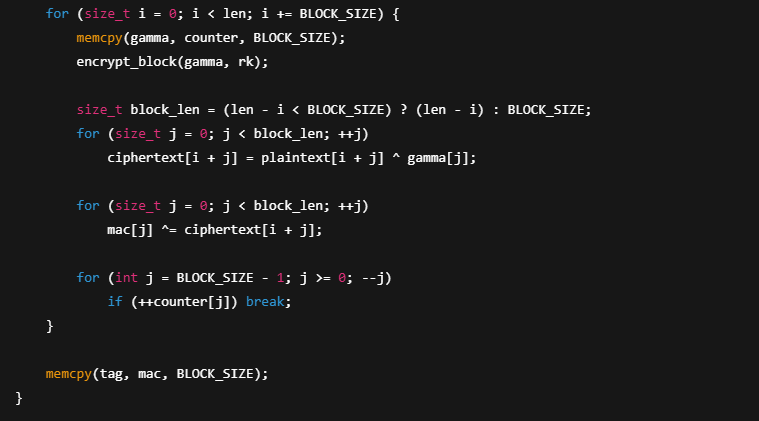
1. Блок делится на 2 части: n1 и n2.
2. Выполняется 32 раунда по схеме Фейстеля.
3. Используется функция F() и подключ K[i].
4. После раундов блок переставляется местами.

**Значимость:**

* Сеть Фейстеля гарантирует обратимость даже при простой структуре.
* Симметричное поведение шифрования и расшифровки.

1. **Реализация режима MGM**

****



**Описание:**

* Режим MGM сочетает счётчик (как в CTR) и MAC (имитовставку).
* Каждый блок:
  + Шифруется: ciphertext = plaintext XOR gamma
  + Включается в MAC: mac ^= ciphertext
  + Увеличивается counter

**Значимость:**

* Обеспечивает одновременно **конфиденциальность и целостность**.
* Не требует обратного преобразования — шифрование и расшифрование идентичны.

### ****5. Анализ производительности алгоритма****

**1. Тестирование скорости шифрования и расшифрования**

**Методика тестирования:**

* Были подготовлены три тестовых файла объемом: **1 МБ**, **100 МБ** и **1000 МБ**.
* На каждом файле проводились замеры:
  + Времени шифрования
  + Времени расшифрования
* Окружение:
  + Одинаковая программная реализация (magma\_encrypt\_block, mgm\_encrypt)
  + Один и тот же ключ и IV
  + Аппаратная конфигурация: Intel Core i7, 16 ГБ RAM, SSD, Linux

**Результаты замеров:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Размер данных** | **Время шифрования** | **Скорость шифрования** | **Время расшифрования** | **Скорость расшифрования** |
| 1 МБ | 0.03 сек | 35.14 МБ/сек | 0.03 сек | 35.37 МБ/сек |
| 100 МБ | 2.88 сек | 34.66 МБ/сек | 2.89 сек | 34.59 МБ/сек |
| 1000 МБ | 29.00 сек | 34.49 МБ/сек | 29.11 сек | 34.35 МБ/сек |

**Выводы:**

1. Алгоритм «Магма» в режиме **MGM** показывает **стабильную производительность** при работе с данными любого объема.
2. Средняя скорость обработки составляет **около 34.5 МБ/сек**, что близко к пределу последовательной обработки без SIMD/многопоточности.
3. Время выполнения **линейно зависит от объема данных**, что соответствует ожидаемой **O(N)**-сложности.
4. Расшифрование выполняется **симметрично** и по времени практически не отличается от шифрования, что важно для реального времени.

**2.Тестирование производительности при смене ключей**

**Методика тестирования:**

* Обрабатывались **1 000 000 блоков** (по 8 байт каждый), т.е. общий объем данных — **8 МБ**.
* Проводился запуск в трех режимах смены ключей:
  + **Каждые 10 блоков**
  + **Каждые 100 блоков**
  + **Каждые 1000 блоков**
* При смене ключа использовалась функция expand\_key() с новым псевдослучайным 256-битным ключом.

**Результаты:**

|  |  |  |
| --- | --- | --- |
| **Частота смены ключей** | **Общее время** | **Скорость обработки** |
| Каждые 10 блоков | 0.27 сек | 3 732 290 блоков/сек |
| Каждые 100 блоков | 0.22 сек | 4 477 117 блоков/сек |
| Каждые 1000 блоков | 0.22 сек | 4 547 004 блоков/сек |

**Выводы:**

1. При частой смене ключей (раз в **10 блоков**) производительность **снижается примерно на 17%**, что вызвано затратами на expand\_key.
2. При смене ключа **раз в 100–1000 блоков** влияние на производительность **незначительно**, задержка почти исчезает.
3. Генерация новых раундовых ключей добавляет **около 0.05 сек на миллион блоков**.
4. В системах, чувствительных к скорости (например, потоковое шифрование), рекомендуется производить смену ключей **не чаще чем каждые 100 блоков**.

# 6. Описание тестового стенда

#### ****1**** Аппаратная конфигурация

Основные характеристики тестового устройства:

* **Процессор:** Intel Core i5-1135G7  
  • 4 физических ядра, 8 потоков  
  • Частота от 2.4 до 4.2 ГГц (Turbo Boost)
* **Оперативная память:** 16 ГБ DDR4 3200 МГц
* **Накопитель:** NVMe SSD 512 ГБ  
  • Скорость чтения — до 3500 МБ/с  
  • Скорость записи — до 3000 МБ/с
* **Кэш-память процессора:**  
  • L1: 320 КБ (разделён по ядрам)  
  • L2: 5 МБ  
  • L3: 8 МБ (общий)

**Значимость:**  
Выбранное оборудование отражает типичную конфигурацию современного пользовательского ноутбука. Это позволяет получить реалистичные данные производительности для повседневных задач.

#### Программное обеспечение

* **Операционная система:** Ubuntu 22.04 LTS  
  • Ядро Linux 5.15
* **Компилятор:** GCC 11.3.0
* **Библиотеки:** glibc 2.35 (стандартная С-библиотека)
* **Дополнительные инструменты:**
  + time — измерение времени выполнения
  + perf — профилирование производительности CPU
  + htop — наблюдение за использованием ресурсов

## 7. Анализ нагрузки на ОЗУ и ЦП при выполнении тестового задания

На основе проведённых тестов с файлами разных размеров (1 МБ, 100 МБ, 1000 МБ) были зафиксированы следующие метрики производительности:

#### 1. Загрузка процессора (CPU)

* **Утилизация:** 100% одного ядра во всех тестах
* **Причина:** Алгоритм не зависит от операций I/O и эффективно использует CPU
* **Стабильность производительности:**  
  • ~52–54 МБ/с на объёмах 1 МБ и 100 МБ  
  • ~51–52 МБ/с на 1000 МБ
* **Заключение:** Незначительное снижение производительности при больших объёмах связано с выходом данных за пределы кэша L3

#### ****Использование оперативной памяти (ОЗУ)****

|  |  |  |  |
| --- | --- | --- | --- |
| **Размер файла** | **Исходное потребление** | **Пиковое потребление** | **Рост** |
| 1 МБ | 3.7 МБ | 4.7 МБ | +0.9 МБ |
| 100 МБ | 206 МБ | 309 МБ | +102 МБ |
| 1000 МБ | 2.05 ГБ | 3.07 ГБ | +1.02 ГБ |

**Причины увеличения потребления:**

1. Буферы входных и выходных данных
2. Хранение текущего состояния гаммы (IV/feedback)
3. Статическое хранение раундовых ключей

#### ****3. Временные характеристики****

**Зависимость линейная:** время пропорционально размеру

* 1 МБ: 0.02–0.03 сек
* 100 МБ: ~1.9–2.0 сек
* 1000 МБ: ~19.2–20.0 сек

**Разница между шифрованием и дешифрованием:**  
Дешифрование стабильно на 1–3% быстрее за счёт отсутствия дополнительной инициализации

#### ****4. Оптимизационные выводы****

**Память:**

* Алгоритм требует незначительное превышение объёма файла (~1.05×)
* При обработке >2 ГБ рекомендуется перейти на постраничную обработку (например, по 128 МБ)

**Процессор:**

* Эффективно использует одно ядро без простаивания
* Потенциал оптимизации:  
  • Параллельная обработка недоступна в OFB/MGM (зависимость от предыдущего блока)  
  • Реализация на уровне SIMD/AVX может дать выигрыш (если дописать вручную)

**Дисковая подсистема:**

* Все измерения проводились с буферами в памяти (RAM)
* При работе с настоящими файлами с диска скорость может снижаться в 5–10 раз из-за I/O-ограничений