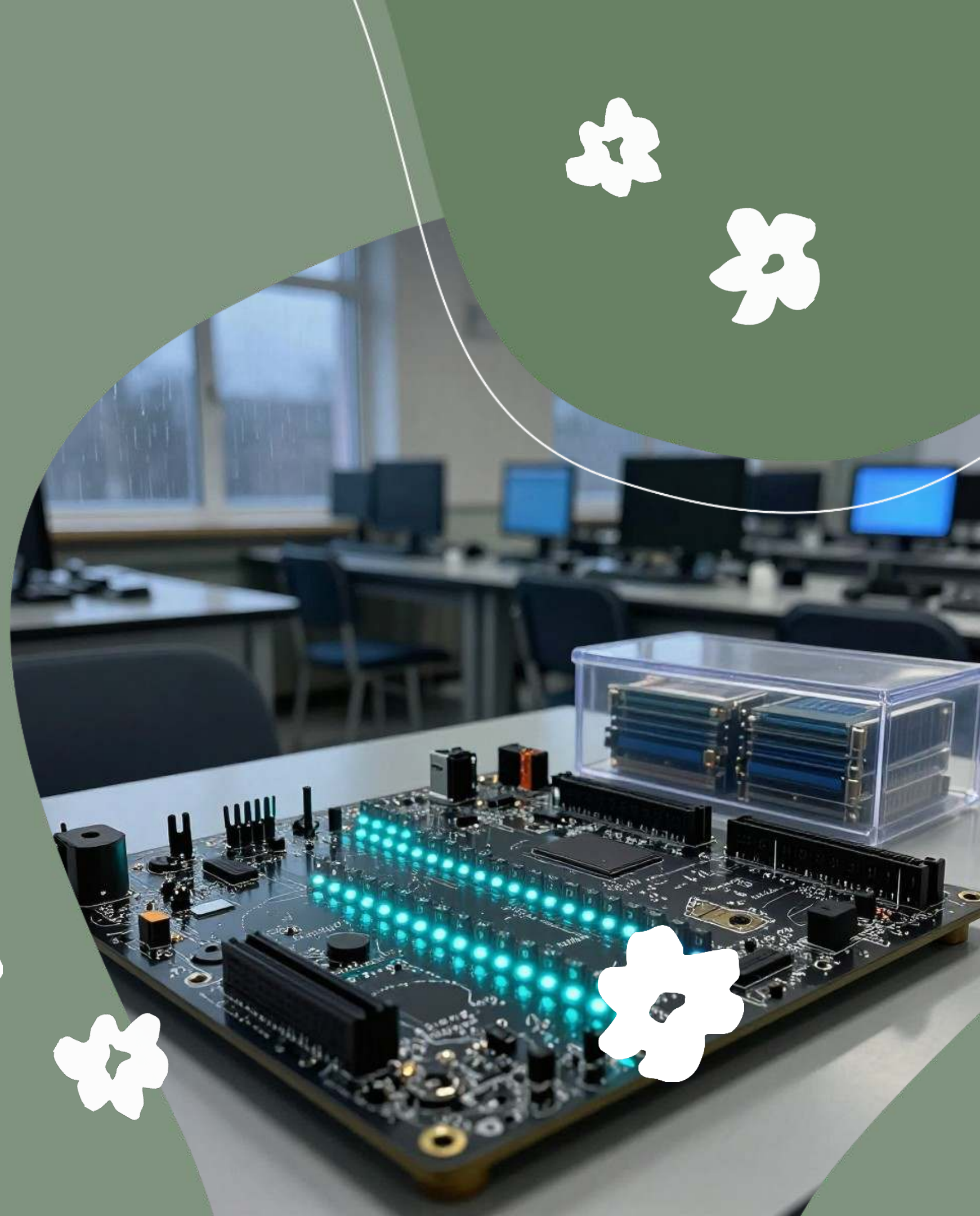


Двоичная система: язык компьютеров и хранение данных

Компьютер хранит всё как цепочки 0 и 1: от текста до видео и программ.



Почему двоичная система стала основой информатики

Двоичная система использует всего два символа — 0 и 1. Такое удобно для электроники: легко различать два устойчивых состояния. Истоки идеи связывают с Лейбницем и Булем, а затем она стала фундаментом цифровой логики и вычислений.



Бит, байт и единицы измерения данных

Бит — минимальная единица информации: он может быть только 0 или 1. Именно из таких «кирпичиков» строятся более сложные данные, которые потом обрабатываются программами.

8 бит составляют 1 байт. Обычно в байте помещается один символ текста, например буква, цифра или знак препинания. Поэтому размер текста можно оценивать по количеству символов.

Для крупных объёмов используют килобайты, мегабайты, гигабайты и терабайты. В школьной информатике часто применяют соотношения $1 \text{ КБ} = 1024 \text{ байта}$, $1 \text{ МБ} = 1024 \text{ КБ}$, $1 \text{ ГБ} = 1024 \text{ МБ}$.

Рост объёма данных по степеням двойки



Соотношения: 1 КБ = 1024 байта; 1 МБ = 1 048 576 байта; 1 ГБ = 1 073 741 824 байта; 1 ТБ = 1 099 511 627 776 байт.

Каждый шаг увеличивает объём в 1024 раза, поэтому график выглядит «ступеньками» и быстро уходит вверх.

Такой рост объясняет, почему даже небольшой носитель вмещает много файлов, а большие медиатеки требуют гигабайты и терабайты.

Как число записывается в двоичном виде

Вес разрядов: почему 13 превращается в 1101

Десятичное число 13 в двоичной системе записывается как 1101. Это связано с весами разрядов 1, 2, 4 и 8: $13 = 8 + 4 + 1$. Так видно, какие «части» числа набраны.

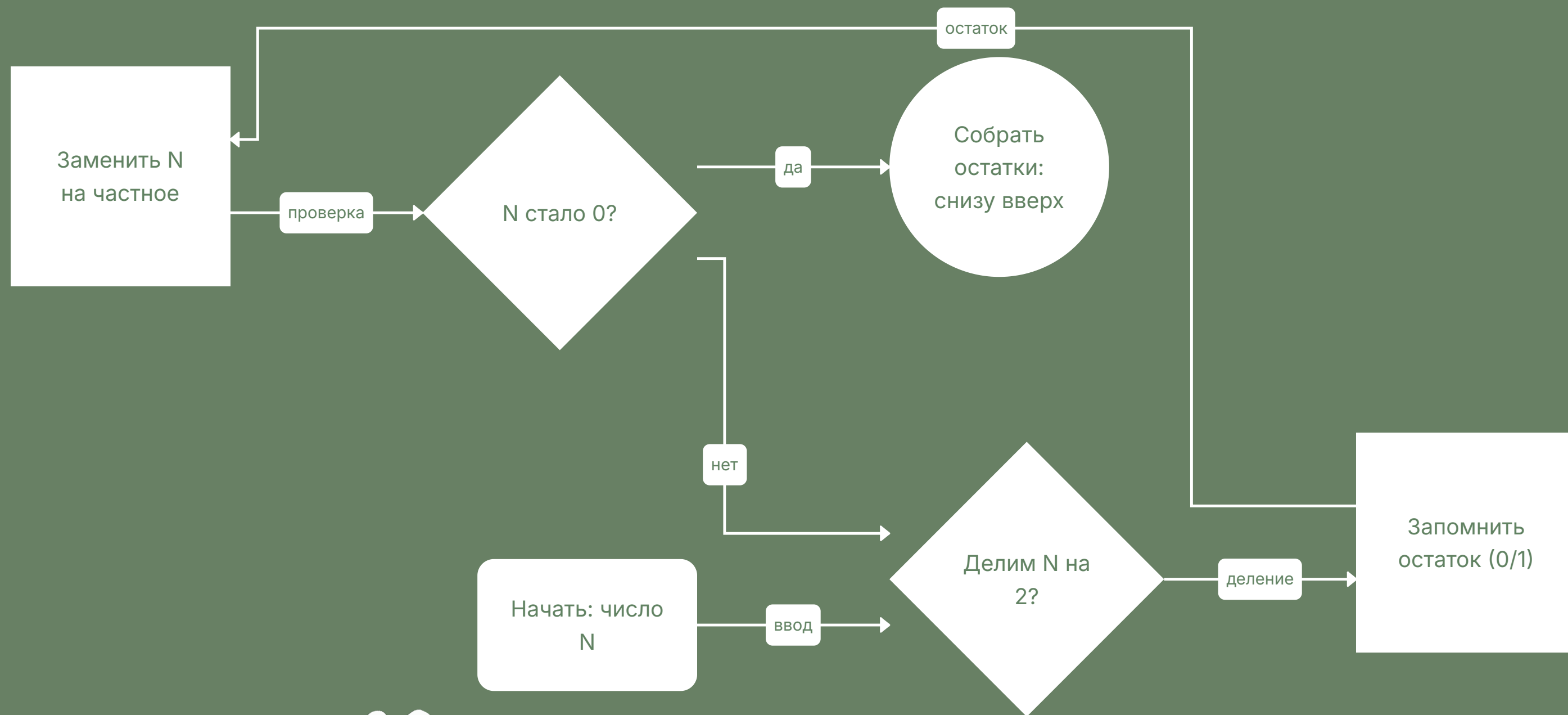
Где пригодятся разные системы счисления

Для сравнения обычно используют три формата записи: десятичную (как привычно человеку), двоичную (как работает компьютер) и дополнительные удобные варианты — восьмеричную или шестнадцатеричную. Их применяют, когда двоичных цифр слишком много.



Перевод числа из десятичной системы в двоичную

Алгоритм основан на последовательном делении на 2 с фиксацией остатков.



Почему компьютеру удобно использовать 0 и 1



Два устойчивых состояния

Электронные схемы надёжно различают два режима: сигнал есть или сигнала нет, высокий уровень или низкий. Поэтому ошибки возникают реже, даже если есть помехи.



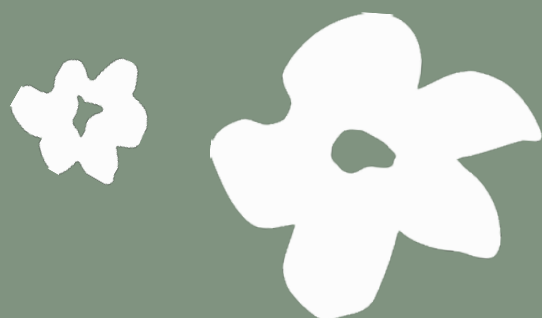
Надёжная обработка данных

Переход между состояниями и хранение информации становятся понятными и повторяемыми. Компьютер проще контролирует, что именно записано и что нужно прочитать дальше.



Логические операции

На двоичных сигналах строятся операции И, ИЛИ и НЕ. Из них процессор собирает вычисления и простые решения, например «если условие верно — делай так».



Основные единицы измерения памяти

Внутренний объём памяти измеряется через степени двойки: при переходе к следующей единице значение увеличивается в 1024 раза.

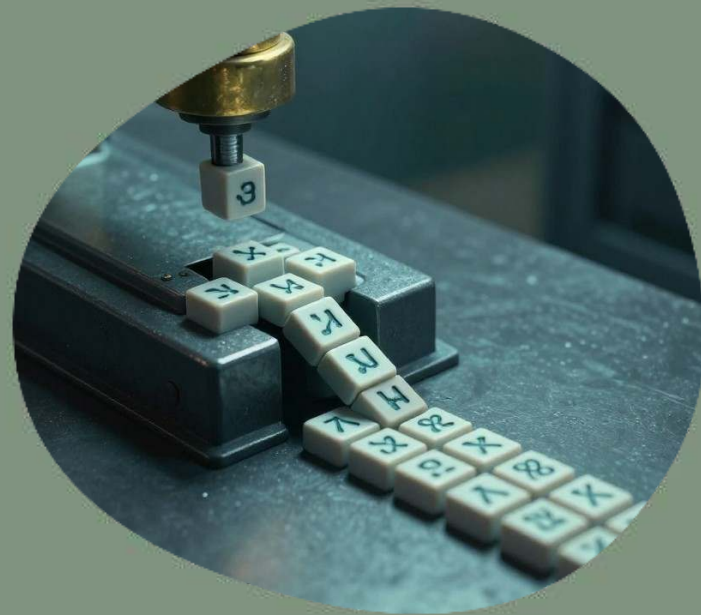
Понимание этих переходов помогает оценивать, сколько данных «помещается» на носителе и почему медиаконтент быстро растёт по объёму.

Ед.	Соотношение / бытовой пример
Бит	1 бит — один сигнал 0 или 1
Байт	1 байт = 8 бит — обычно 1 символ текста
Килобайт	1 КБ = 1024 байта — небольшой фрагмент текста
Мегабайт	1 МБ = 1024 КБ — фотография или часть аудио
Гигабайт	1 ГБ = 1024 МБ — фильм/большая папка файлов
Терабайт	1 ТБ = 1024 ГБ — обширная коллекция видео и документов

Соотношения даны в школьной информатике: 1 КБ = 1024 байта, 1 МБ = 1024 КБ, 1 ГБ = 1024 МБ, 1 ТБ = 1024 ГБ.



Кодирование текста в компьютере



Когда буква становится числом

Текст на экране хранится как последовательность числовых кодов. Компьютер сначала выбирает код для каждого символа, затем записывает этот код в двоичном виде. Поэтому один и тот же текст можно передавать и сохранять одинаково на разных устройствах.



ASCII: базовый набор символов

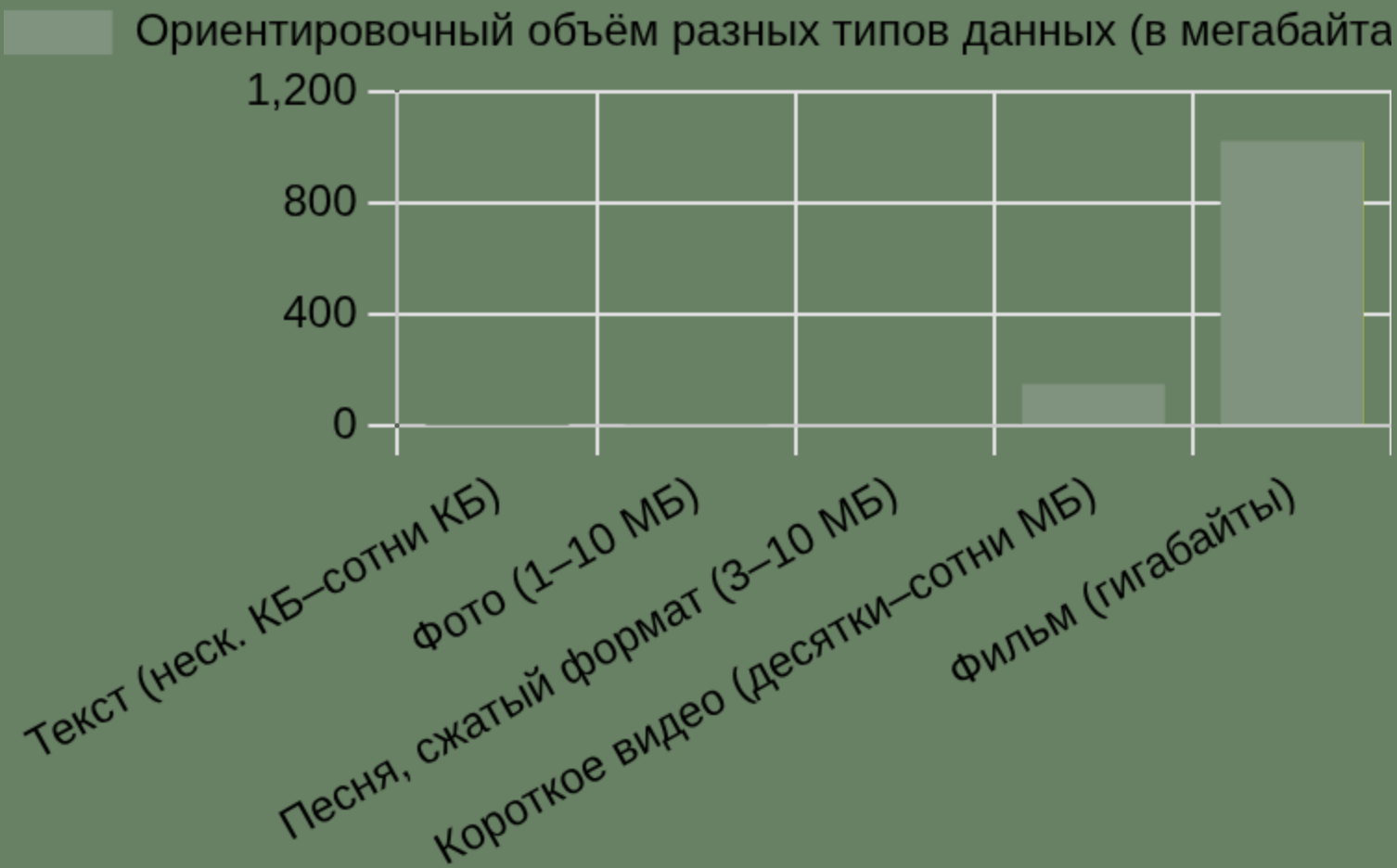
В ASCII для базового набора используют 7 бит, что даёт 128 возможных символов. Этого достаточно для латиницы, цифр и многих знаков. Но для всех языков мира одного набора кодов мало — нужен более широкий стандарт.



Unicode: один код для многих языков

Unicode расширяет систему так, что поддерживаются символы разных языков, математические знаки и специальные символы. Поэтому одна и та же буква может получить код, а затем — двоичную запись, независимо от того, латиница это или кириллица.

Сравнение объёма текста, фото, звука и видео



Оценки из описания: текст — от нескольких КБ до сотен КБ; фото — 1–10 МБ; песня — около 3–10 МБ; короткое видео — десятки/сотни МБ; фильм — гигабайты.

Видео чаще всего больше всего, потому что в нём много кадров и каждый кадр несёт картинку, а рядом ещё хранится звук.

Разные типы файлов занимают разный объём из-за количества «закодированных элементов»: символов, пикселей, измерений звука и кадров.

Как изображение хранится в памяти

Изображение на экране состоит из пикселей — маленьких точек. Каждый пиксель соответствует одному набору битов, который хранит код цвета. Поэтому картинка — это не «рисунок», а набор чисел.

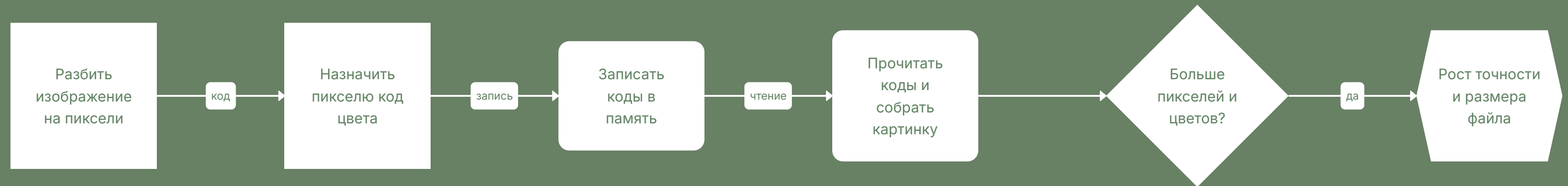
Разрешение определяется количеством пикселей по ширине и высоте. Чем их больше, тем точнее передаются детали: контуры становятся резче, мелкие объекты различимее. При росте пикселей увеличивается и объём данных.

Помимо количества пикселей, важны и доступные цвета. Если используется больше оттенков, каждому пикселю нужно больше бит для точного кодирования. Поэтому при «богатой палитре» файл становится тяжелее.

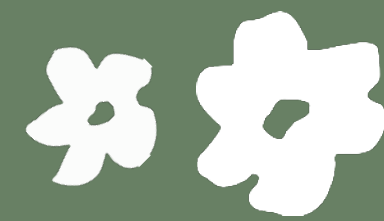
Фотографии с камеры обычно занимают больше места, чем простые значки: они содержат миллионы пикселей и много оттенков. В итоге изображение кодируется детально, и на хранение требуется больше байтов.

Как компьютер сохраняет и восстанавливает изображение

Процесс преобразует картинку в набор кодов цветов, который хранится в памяти, а затем обратно собирается на экране. Чем больше пикселей и цветов, тем больше данных.



Звук и видео как последовательности битов



1. Дискретизация звука

Звук измеряют через равные промежутки времени: получается ряд значений громкости. Каждое значение превращается в число и записывается как последовательность битов.

2. Кодирование качества

Если увеличить частоту дискретизации и глубину кодирования, измерения становятся точнее. Но тогда на каждую секунду требуется больше бит, и файл растёт в объёме.

3. Кадры видеопотока

Видео состоит из множества кадров, сменяющихся с частотой, например 24, 30 или 60 раз в секунду. Каждый кадр кодируется так же, как изображение, пиксель за пикселем.

4. Итоговый объём файла

Поскольку видео хранит много кадров и обычно ещё звуковую дорожку, оно почти всегда больше, чем текст, фото или аудио. Размер зависит от настроек кодирования.



Почему разные типы файлов занимают разный объём

Объём файла определяется тем, сколько элементов нужно закодировать: символов для текста, пикселей для изображений, измерений для звука и кадров для видео. Поэтому размеры так различаются на практике.

Файл становится больше, когда требуется закодировать больше элементов: больше пикселей, больше временных измерений или больше кадров, а также при менее эффективном сжатии.

Тип файла	Пример	Почему отличается
Текст	сообщение, конспект	Мало символов или простая структура: кодируется небольшой набор знаков.
Фото	картинка с камеры	Много пикселей и цветов: нужен подробный код для каждого элемента.
Аудио	песня	Зависит от длительности, частоты дискретизации и качества сжатия: больше измерений — больше данных.
Видео	фильм, ролик	Большое число кадров и часто звук: кодируются одновременно изображения и аудиодорожка.

Обобщение принципов кодирования данных (пиксели, измерения, кадры).

Где в компьютере хранятся биты и байты

- Оперативная память хранит данные временно и быстро: процессор обращается к ней, чтобы читать и записывать результаты вычислений. Когда программа заканчивается, содержимое ОЗУ может исчезнуть.
- Накопители SSD или HDD сохраняют файлы надолго. Файл — упорядоченная последовательность байтов, а папки помогают организовать много файлов. На практике процессор постоянно «обменивается» данными с памятью и накопителем.



Как быстро растёт размер файлов



Приблизжённые ориентиры из школьной практики и типичного сжатия; точные значения зависят от формата и настроек.

Переход к мультимедиа увеличивает объём не «в раз» по одному параметру, а сразу по нескольким: больше пикселей, больше измерений во времени и больше информации на секунду.

Чем больше данных нужно закодировать для каждой секунды и для каждого кадра, тем быстрее растёт размер. Поэтому видео чаще всего оказывается самым тяжёлым типом файла.

Ошибки при хранении данных и защита информации

Резервные копии



Если файл повреждён, резервная копия даёт возможность восстановить прежнюю версию. Копирование особенно важно для фотографий, школьных проектов и сообщений, которые нельзя потерять.

Каналы, где возникают ошибки



Биты могут искажаться из-за сбоев в памяти, повреждения носителя, помех при передаче или внезапного отключения питания. Такие ситуации нарушают «правильные» нули и единицы.

Контрольные суммы



Контрольная сумма вычисляется по содержимому файла и помогает проверить целостность. При ошибке она не совпадает, и система понимает, что данные могли исказиться при записи или передаче.

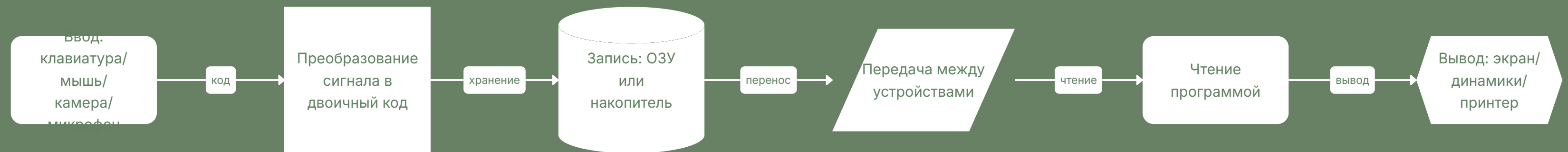
Обнаружение и исправление



Коды обнаружения и исправления ошибок помогают распознать проблему и иногда восстановить данные. Это снижает риск потерять информацию и повышает надёжность хранения и передачи.

Путь данных от ввода до отображения

Информация проходит цепочку: входные сигналы превращаются в биты, которые хранятся и обрабатываются, затем выводятся устройствами. На каждом шаге меняется представление, но остаются биты.



Двоичный код в повседневной цифровой среде

QR-коды, номера телефонов, файлы на флешке и сообщения в мессенджерах передаются и хранятся одинаково по смыслу: всё сводится к битам и байтам. Устройствам важно лишь уметь кодировать и читать их.

Фотографии в облаке и игры на смартфоне тоже опираются на те же базовые правила. Поэтому одна и та же информация может открываться на разных устройствах, если они используют согласованные форматы.

Двоичная система лежит в основе работы интернета, компьютерных сетей и большинства электронных устройств. В каждом «переходе» по цепочке данные остаются последовательностью 0 и 1, меняется лишь способ обработки.

Главный вывод: почему двоичная система важна для информатики

Компьютер хранит и обрабатывает любые данные как последовательности 0 и 1. Биты складываются в байты, байты — в файлы, а файлы превращаются в тексты, изображения, звук и видео, делая цифровые устройства понятными и управляемыми.