**Типы данных, переменные**

**Переменная** - это ячейка в оперативной памяти  микроконтроллера, которая имеет своё уникальное название (а также адрес в памяти) и хранит значение соответственно своему размеру. К переменной мы можем обратиться по её имени или адресу и получить это значение, либо изменить его. Зачем это нужно? В переменной могут храниться промежуточные результаты вычислений, полученные "снаружи" данные (с датчиков, Интернета, интерфейсов связи) и так далее.

## Измерение информации

Прежде чем перейти к переменным и их типам, нужно вспомнить школьный курс информатики, а именно - как хранятся данные в "цифровом" мире. Любая память состоит из элементарных ячеек, которые имеют всего **два** состояния: 0 и 1. Эта единица информации называется **бит**(**bit**). Минимальным блоком памяти, к которому можно обратиться из программы по имени или адресу, является **байт** (**byte**), который в Arduino (и в большинстве других платформ и процессоров) состоит из **8 бит**, таким образом любой тип данных будет кратен 1 байту.

Максимальное количество значений, которое можно записать в один байт, составляет 2^8 = 256. В программировании счёт всегда начинается с нуля, поэтому один байт может хранить число от **0 до 255**. Более подробно о двоичном представлении информации и битовых операциях мы поговорим в отдельном уроке.

Стандартные типы переменных в Arduino по своему размеру кратны степени двойки, давайте их распишем:

* 1 байт = 8 бит = 256
* 2 байта = 16 бит = 65 536
* 4 байта = 32 бита = 4 294 967 296

## Типы данных

Переменные разных типов имеют разные особенности и позволяют хранить числа в разных диапазонах.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Название** | **Альт. название** | **Вес** | **Диапазон** | **Особенность** |
| boolean | bool | 1 байт **\*** | 0 или 1,**true**  или **false** | Логический тип |
| char | - | 1 байт | -128… 127 (AVR), 0.. 255 (esp) | Символ (код символа) из таблицы ASCII |
| - | int8\_t | 1 байт | -128… 127 | Целые числа |
| byte | uint8\_t | 1 байт | 0… 255 | Целые числа |
| int **\*\*** | int16\_t, short | 2 байта | -32 768… 32 767 | Целые числа. На ESP8266/ESP32 - 4 байта! См. ниже |
| unsigned int**\*\*** | uint16\_t  , word | 2 байта | 0… 65 535 | Целые числа. На ESP8266/ESP32 - 4 байта! См. ниже |
| long | int32\_t | 4 байта | -2 147 483 648…    2 147 483 647 | Целые числа |
| unsigned long | uint32\_t | 4 байта | 0… 4 294 967 295 | Целые числа |
| float | - | 4 байта | -3.4E+38… 3.4E+38 | Числа с плавающей точкой, точность: 6-7 знаков |
| double | - | 4/8 байт | -1.7E+308.. 1.7E+308 | Для AVR то же самое, что  float.  На ESP и прочих 32-бит МК - 8 байт, точность - 15-16 знаков |
| - | int64\_t | 8 байт **\*\*\*** | -(2^64)/2... (2^64)/2-1 | Целые числа |
| - | uint64\_t | 8 байт **\*\*\*** | 2^64-1 | Целые числа |

* (**\***) - да,

bool

 занимает 1 байт (8 бит), так как это минимальная адресуемая ячейка памяти. Есть способы запаковать логические переменные в 1 бит, о них поговорим в [другом уроке](https://alexgyver.ru/lessons/code-optimisation/).

* (**\*\***) - на ESP8266/ESP32

int

 и

unsigned int

 занимает 4 байта, то есть является аналогами типов

long

 и

unsigned long

!

* (**\*\*\***) - Компилятор также поддерживает **64 битные** числа. Стандартные Arduino-библиотеки с переменными этого типа не работают, поэтому можно использовать только в своём коде.

**Целочисленные типы**

Переменные целочисленных типов нужны для хранения целых чисел. В своей программе рекомендуется использовать альтернативное название типов (второй столбец в таблице выше), потому что:

* Проще ориентироваться в максимальных значениях
* Легче запомнить
* Название более короткое
* Проще изменить один тип на другой
* Размер переменной задан жёстко и не зависит от платформы (например int

 на AVR это 2 байта, а на esp8266 - 4 байта)

Максимальные значения хранятся в константах, которые можно использовать в коде. Иногда это помогает избавиться от лишних вычислений:

* UINT8\_MAX

 - 255

* INT8\_MAX

 - 127

* UINT16\_MAX

 - 65 535

* INT16\_MAX

 - 32 767

* UINT32\_MAX

- 4 294 967 295

* INT32\_MAX

 - 2 147 483 647

* UINT64\_MAX

 - 18 446 744 073 709 551 615

* INT64\_MAX

 - ‭9 223 372 036 854 775 807

**Логический тип**

bool

 - логический, он же **булевый** (придуман [Джорджем Булем](https://ru.wikipedia.org/wiki/%D0%91%D1%83%D0%BB%D1%8C,_%D0%94%D0%B6%D0%BE%D1%80%D0%B4%D0%B6)) тип данных, принимает значения

0

 и

1

 или

**false**

 и

**true**

 - ложь и правда. Используется для хранения состояний, например включено/выключено, а также для работы в [условных конструкциях](https://alexgyver.ru/lessons/conditions/).

Также переменная типа

bool

 принимает значение

**true**

, если присвоить ей любое **отличное от нуля**число.

bool a = 0; // false

bool b = 1; // true

bool c = 25; // true