

Шпаргалка по Arduino

Создано на основе: Arduino Language Reference
<https://arduino.cc/en/Reference/>

Структуры и потоки

Основная структура программы

```
void setup() {  
    // Выполняется один раз при старте  
}  
void loop() {  
    // Выполняется периодически  
}
```

Структура управления

```
if (x < 5) { ... } else { ... }  
while (x < 5) { ... }  
for (int i = 0; i < 10; i++) { ... }  
break;    // Выйти из цикла немедленно  
continue; // Перейти к следующей итерации  
switch (var) {  
    case 1:  
        ...  
        break;  
    case 2:  
        ...  
        break;  
    default:  
        ...  
}  
return x; // x must match return type  
return;  // For void return type
```

Объявление функций

```
<возвр. тип> <имя>(<параметры>) { ... }  
пример: int double(int x) {return x*2;}
```

Операторы

Основные операторы

= присвоение
+ сложение - вычитание
* умножение / деление
% остаток от деления
== равно != не равно
< меньше чем > больше чем
<= больше или равно
>= меньше или равно
&& И || ИЛИ
! НЕ

Составные операторы

++ инкремент
-- декремент
+= сложение и присвоение
-= вычитание и присвоение
*= умножение и присвоение
/= деление и присвоение
&= логическое И и присвоение
|= логическое ИЛИ и присвоение

Побитовые операторы

& побитовое И | побитовое или
^ побитовое искл.ИЛИ ~ побитовое НЕ
<< сдвиг влево >> сдвиг вправо

Доступ с указателем

& reference: получить указатель
* dereference: перейти к указателю

Встроенные Функции

Порты Ввода/Вывода

Цифровой Вх/Вых - pins 0-13 A0-A5
pinMode(pin,
[INPUT, OUTPUT, INPUT_PULLUP])
int digitalRead(pin)
digitalWrite(pin, [HIGH, LOW])

Аналоговый Вход - pins A0-A5

int analogRead(pin)
analogReference(
[DEFAULT, INTERNAL, EXTERNAL])

ШИМ Выход - pins 3 5 6 9 10 11

analogWrite(pin, value)

Дополнительные Вх/Вых

tone(pin, freq_Hz)
tone(pin, freq_Hz, duration_ms)
noTone(pin)
shiftOut(dataPin, clockPin,
[MSBFIRST, LSBFIRST], value)
unsigned long pulseIn(pin,
[HIGH, LOW])

Время

unsigned long millis()
// Overflows at 50 days
unsigned long micros()
// Overflows at 70 minutes
delay(msec)
delayMicroseconds(usec)

Математические

min(x, y) max(x, y) abs(x)
sin(rad) cos(rad) tan(rad)
sqrt(x) pow(base, exponent)
constrain(x, minval, maxval)
map(val, fromL, fromH, toL, toH)

Случайные числа
randomSeed(seed) // long or int
long random(max) // 0 to max-1
long random(min, max)

Биты и Байты

lowByte(x) highByte(x)
bitRead(x, bitn)
bitWrite(x, bitn, bit)
bitSet(x, bitn)
bitClear(x, bitn)
bit(bitn) // bitn: 0=LSB 7=MSB

Преобразование Типов

char(val) byte(val)
int(val) word(val)
long(val) float(val)

Внешние прерывания

attachInterrupt(interrupt, func,
[LOW, CHANGE, RISING, FALLING])
detachInterrupt(interrupt)
interrupts()
noInterrupts()

Библиотеки

Serial.h - связь с ПК через пины RX/TX
begin(long speed) // Ск-ть до 115200
end()
int available() //к-во байт доступно
int read() // -1 если недоступно
int peek() // Чтение без удаления
flush()
print(data) println(data)
write(byte) write(char * string)
write(byte * data, size)
SerialEvent() // Вызов. если готово

SoftwareSerial.h - через любой пин
SoftwareSerial(rxPin, txPin)
begin(long speed) // Ск-ть д 115200
listen() // Только 1 слуш-ся
isListening() // за период.
read, peek, print, println, write
//Эквивалентно библиотеке Serial

EEPROM.h - Доступ к энергонез. памяти
byte read(addr)
write(addr, byte)
EEPROM[index] // Доступ как массив

Servo.h - управление серво двиг-ми
attach(pin, [min_uS, max_uS])
write(angle) // 0 to 180
writeMicroseconds(uS)
// 1000-2000; 1500 - середина
int read() // 0 to 180 градусов
bool attached()
detach()

Wire.h - I²C подключение
begin() // Подкл. как master
begin(addr) // как slave @ addr
requestFrom(address, count)
beginTransmission(addr) // Шаг 1
send(byte) // Шаг 2
send(char * string)
send(byte * data, size)
endTransmission() // Шаг 3
int available() //к-во байт доступно
byte receive() //получить след. байт
onReceive(handler)
onRequest(handler)

Переменные, Массивы и Данные

Типы данных

boolean true | false
char -128 - 127, 'a' '\$' etc.
unsigned char 0 - 255
byte 0 - 255
int -32768 - 32767
unsigned int 0 - 65535
word 0 - 65535
long -2147483648 - 2147483647
unsigned long 0 - 4294967295
float -3.4028e+38 - 3.4028e+38
double то же самое, что и float
void т.е. без возврата значения

Строки

```
char str1[8] =  
{ 'A', 'r', 'd', 'u', 'i', 'n', 'o', '\0' };  
// Включая \0 пустой возврат  
char str2[8] =  
{ 'A', 'r', 'd', 'u', 'i', 'n', 'o' };  
// Компилятор добавляет пустой возврат  
char str3[] = "Arduino";  
char str4[8] = "Arduino";
```

Числовые константы

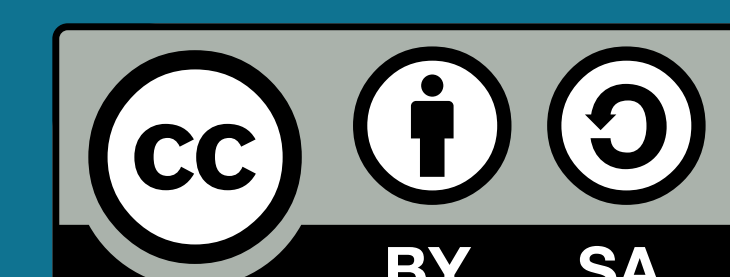
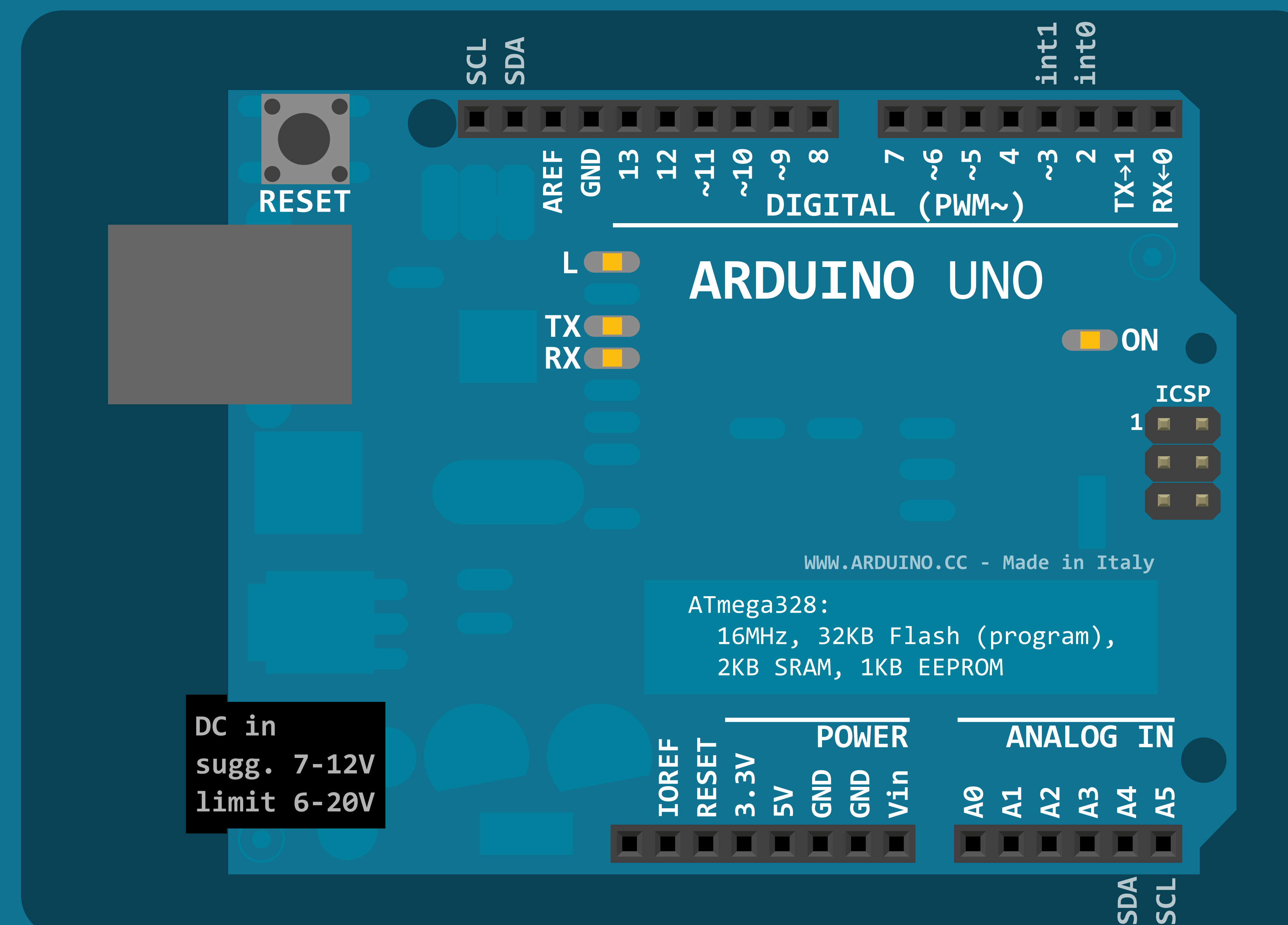
123 десятичное
0b01111011 двоичное
0173 восьмиричное - осн. 8
0x7B шестнадцатичное-осн. 16
123U force unsigned
123L force long
123UL force unsigned long
123.0 force floating point
1.23e6 1.23*10^6 = 1230000

Спецификаторы

static persists between calls
volatile в ОЗУ (nice for ISR)
const только для чтения
PROGMEM во флеш памяти

Массивы

```
int myPins[] = {2, 4, 8, 3, 6};  
int myInts[6]; // Array of 6 ints  
myInts[0] = 42; // Assigning first  
// index of myInts  
myInts[6] = 12; // ERROR! Indexes  
// are 0 though 5
```



by Mark Liffiton
version: 2017-04-10

перевод: Яцемирский Андрей

Источник: <https://github.com/liffiton/Arduino-Cheat-Sheet/>

Адаптация:

- Оригинал: Gavin Smith
- SVG версия: Frederic Dufourg
- Рисунок платы Arduino: Fritzing.org